

OPTIMASI KOMPOSISI PAKAN TERNAK AYAM PETELUR MENGUNAKAN ALGORITME GENETIKA

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Siti Fatimah Al Uswah
NIM: 115060801111064



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2018**

OPTIMASI KOMPOSISI PAKAN TERNAK AYAM PETELUR MENGUNAKAN ALGORITME GENETIKA

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Siti Fatimah Al Uswah
NIM: 115060801111064



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2018**

PENGESAHAN

OPTIMASI KOMPOSISI PAKAN TERNAK AYAM PETELUR MENGGUNAKAN
ALGORITME GENETIKA

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Siti Fatimah Al Uswah
NIM: 115060801111064

Skripsi ini telah diuji dan dinyatakan lulus pada
3 Agustus 2018

Telah diperiksa dan disetujui oleh:

Pembimbing I

Pembimbing II

Budi Dharma Setiawan, S.Kom., M.Cs
NIP: 198410152014041002

Dian Eka Ratnawati, S.Si., M.Kom
NIP: 19730619 200212 2 001

Mengetahui
Ketua Jurusan Teknik Informatika

Tri Astoto Kurniawan, S.T., M.T., Ph.D
NIP: 19710518 200312 1 001

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsurunsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 18 Juli 2018

Siti Fatimah Al Uswah
NIM: 115060801111064



KATA PENGANTAR

Puji syukur kehadiran Allah SWT yang telah melimpahkan rahmat, taufik dan hidayah-Nya sehingga laporan skripsi yang berjudul “ **Optimasi Komposisi Pakan Ternak Ayam Petelur Menggunakan Algoritme Genetika**” ini dapat terselesaikan.

Penulis menyadari bahwa skripsi ini tidak akan berhasil tanpa bantuan dari beberapa pihak. Oleh karena itu, penulis ingin menyampaikan rasa hormat dan terima kasih kepada:

1. Budi Darma Setiawan, S.Kom., M.Cs. selaku dosen pembimbing 1 dan Dian Eka Ratnawati, S.Si, M.Kom. selaku dosen pembimbing 2 yang telah dengan sabar membimbing dan mengarahkan penulis sehingga dapat menyelesaikan skripsi ini,
2. Wayan Firdaus Mahmudy, S.Si., M.T., Ph.D., Heru Nurwarsito, Ir., M.Kom., Suprpto, S.T., M.T., Edy Santoso, S.Si., M.Kom. selaku Dekan, Wakil Dekan I, Wakil Dekan II, dan Wakil Dekan III Fakultas Ilmu Komputer Universitas Brawijaya,
3. Agus Wahyu Widodo, S.T., M.Cs. selaku Ketua Program Studi Teknik Informatika,
4. Tri Astoto Kurniawan, S.T., M.T., Ph.D. selaku Ketua Jurusan Teknik Informatika,
5. Budi Darma Setiawan, S.Kom., M.Cs. selaku dosen Penasihat Akademik yang selalu memberikan nasehat kepada penulis selama menempuh masa studi,
6. Suparlan dan Tetik Eryamti Herlinawati selaku kedua orangtua penulis yang selaku mendukung baik secara moril maupun materil dari awal pendidikan hingga menyelesaikan skripsi ini,
7. Abi, Umi, dan seluruh keluarga besar atas segala nasihat, kasih sayang, perhatian dan kesabarannya di dalam membesarkan dan mendidik penulis, serta yang senantiasa tiada henti-hentinya memberikan doa dan semangat demi terselesaikannya skripsi ini,
8. Kakak Annisau Istighotsah, Adik Sayyid Ridho, Adik Nur Na'imah, dan Adik Yumna Salsabila yang tiada lelah selalu mengingatkan penulis untuk segera wisuda,
9. Arinda Hapsari Achnas, Evi Nur Azizah, Nurbaiti Wahid, dan Riska Amalia Praptiwi selaku teman-teman seperjuangan yang banyak memberikan waktu, dukungan, dan bantuan kepada penulis selama menempuh pendidikan di Universitas Brawijaya,
10. Teman-teman di Ar-Rohmah Putri yang telah banyak memberi bantuan dan dukungan selama penulis menempuh studi di Informatika Universitas Brawijaya dan selama penyelesaian skripsi ini,
11. Dr. M. Halim Natsir, S.Pt., MP. Dosen selaku pakar nutrisi dan kandungan pakan ternak dari Fakultas Peternakan Universitas Brawijaya yang telah membantu kami memberikan data untuk skripsi ini dan Mellinda Ajeng

Jayanti, S.Kom yang telah membantu penulis dalam menghubungkan penulis ke pakar,

12. Seluruh civitas akademika Informatika Universitas Brawijaya yang telah banyak memberi bantuan dan dukungan selama penulis menempuh studi di Informatika Universitas Brawijaya dan selama penyelesaian skripsi ini.

Penulis menyadari bahwa dalam penyusunan skripsi ini masih banyak kekurangan, sehingga saran dan kritik yang membangun sangat penulis harapkan. Akhir kata penulis berharap skripsi ini dapat membawa manfaat bagi semua pihak yang menggunakannya.

Malang, 18 Juli 2018

Siti Fatimah Al Uswah

Email: sfaluswah@student.ub.ac.id



ABSTRAK

Siti Fatimah Al Uswah, Optimasi Komposisi Pakan Ternak Ayam Petelur Menggunakan Algoritme Genetika

Pembimbing: Budi Darma Setiawan, S.Kom., M.Cs. dan Dian Eka Ratnawati, S.Si., M.Kom.

Peluang beternak ayam petelur dinilai cukup menjanjikan di Indonesia karena Kebutuhan telur dalam negeri terus meningkat sejalan dengan peningkatan pola hidup manusia dalam meningkatkan kebutuhan akan protein hewani. Berdasarkan data dari kementerian pertanian pada tahun 2017 terdapat peningkatan konsumsi telur ayam ras selama tahun 1987-2017 sebesar 3,57% pertahun dengan rata-rata konsumsi 6,63 kg/kap/th pada tahun 2017. Namun dalam beternak ayam petelur biaya pembelian pakan ternak dapat mencapai 60%-70 % dari biaya produksi. Salah satu cara agar dapat menekan biaya pembelian pakan yaitu dengan melakukan optimasi komposisi pakan sehingga dapat diperoleh komposisi pakan yang memenuhi kebutuhan nutrisi dengan biaya yang minimal.

Metode optimasi yang digunakan pada penelitian ini yaitu algoritme genetika dengan representasi permutasi, *single-point crossover*, *reciprocal exchange mutation*, dan seleksi *elitism*. Pada penelitian ini digunakan 50 data bahan pakan ayam petelur beserta kandungan nutrisinya. Dari hasil pengujian yang dilakukan diperoleh parameter populasi dengan nilai fitness tertinggi pada populasi 500 dan 800 dengan rata-rata nilai *fitness* 2.573591, Generasi optimal sebanyak 100 generasi dengan rata-rata nilai *fitness* 2.479726 dan kombinasi probabilitas *crossover* 0.5 dan probabilitas mutasi 0.3 dengan rata-rata nilai *fitness* 2.58459. Hasil akhir yang didapatkan berupa komposisi pakan ternak ayam petelur yang memenuhi kebutuhan nutrisi dengan biaya yang minimal.

Kata kunci: Optimasi komposisi, ayam petelur, algoritme genetika.

ABSTRACT

Siti Fatimah Al Uswah, *Feed Composition Optimization of Laying chicken Using Genetic Algorithm*

Advisor: Budi Darma Setiawan, S.Kom, M.Cs. and Dian Eka Ratnawati, S.Si, M.Kom.

Raising laying hens are considered a promising opportunity in Indonesia because the demand for eggs in the country continues to increase in line with the increasing human lifestyle and need for animal protein. Based on data from the ministry of agriculture in 2017 there is an increase in chicken egg consumption during the year 1987-2017 of 3.57% per year with an average consumption of 6.63 kg / kap / th in 2017. On the other hand, raising laying hens is costly especially when it comes to livestock feed, which can cost farmers 60% -70% of production costs. One way to reduce the cost of purchasing feed is by optimizing the feed composition, with purpose of achieving an optimal feed composition that also meets the nutritional needs, all obtained with as minimal cost as possible.

The optimization method used in this research is Genetic Algorithm with permutation representation, single-point crossover, reciprocal exchange mutation, and elitism selection. This study used 50 feed data material of laying chicken and its nutritional content. From the results of the tests, the population parameters obtained with the highest fitness value in the population of 500 and 800 with the average fitness value of 2.573591, the optimal generation of 100 generations with an average fitness value of 2.479726 and a combination of probability of crossover 0.5 and the probability of mutation 0.3 with the average fitness value 2.58459. The final result is the composition of laying chicken feed that meets the nutritional needs with minimal cost.

Keywords: Optimization of composition, laying hens, genetic algorithm.

DAFTAR ISI

OPTIMASI KOMPOSISI PAKAN TERNAK AYAM PETELUR MENGGUNAKAN ALGORITME GENETIKA.....	i
PERSETUJUAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	vi
ABSTRACT.....	vii
DAFTAR ISI.....	viii
DAFTAR TABEL.....	xi
DAFTAR GAMBAR.....	xii
DAFTAR SOURCE CODE	xiii
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan	3
1.4 Batasan Masalah	3
1.5 Manfaat	4
1.6 Sistematika Penulisan.....	4
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Kajian Teori.....	5
2.2 Ayam Petelur.....	7
2.2.1 Sejarah Ayam Petelur.....	8
2.2.2 Fase Ayam Petelur.....	8
2.2.3 Jenis-jenis Ayam Petelur	9
2.3 Nutrisi Pakan.....	10
2.3.1 Energi	10
2.3.2 Protein.....	11
2.3.3 Lemak	11
2.3.4 Mineral	12
2.4 Algoritme Genetika.....	13

2.4.1	Kelebihan Algoritme Genetika	14
2.4.2	Struktur Umum Algoritme Genetika	14
2.4.3	Komponen Utama Algoritme Genetika.....	16
2.4.4	Kondisi Berhenti (<i>Termination Condition</i>)	19
BAB 3 METODOLOGI		20
3.1	Tahapan Penelitian	20
3.2	Studi Literatur	21
3.3	Pengumpulan Data	21
3.4	Analisa dan Perancangan	21
3.4.1	Analisa Kebutuhan	21
3.4.2	Deskripsi Permasalahan	22
3.4.3	Deskripsi Umum Sistem	24
3.4.4	Proses Perhitungan	25
3.5	Implementasi.....	40
3.6	Perancangan Pengujian	40
BAB 4 PERANCANGAN.....		41
4.1	Perancangan Antarmuka	41
4.1.1	Halaman awal.....	41
4.1.2	Halaman inisialisasi algoritme genetika	42
4.1.3	Halaman <i>crossover</i> dan mutasi	43
4.1.4	Halaman evaluasi	44
4.1.5	Halaman seleksi.....	44
4.2	Perancangan Pengujian	45
4.2.1	Uji Coba Banyaknya Generasi	46
4.2.2	Uji Coba Ukuran Populasi.....	46
4.2.3	Uji Coba Kombinasi P_c Dan P_m	47
BAB 5 IMPLEMENTASI		48
5.1	Spesifikasi Aplikasi	48
5.1.1	Spesifikasi Perangkat Keras.....	48
5.1.2	Spesifikasi Perangkat Lunak	48

5.2	Batasan Implementasi	48
5.3	Implementasi Algoritme	49
5.3.1	Implementasi Proses Pembangkitan Populasi Awal	49
5.3.2	Implementasi Proses <i>Crossover</i>	50
5.3.3	Implementasi Proses Mutasi	51
5.3.4	Implementasi Proses Perhitungan Penalti	52
5.3.5	Implementasi Proses Perhitungan <i>Fitness</i>	54
5.3.6	Implementasi proses seleksi	55
5.4	Implementasi Antarmuka	56
5.4.1	Halaman Awal	56
5.4.2	Halaman Inisialisasi	56
5.4.3	Halaman <i>Crossover</i> dan Mutasi	57
5.4.4	Halaman Evaluasi	57
5.4.5	Halaman Seleksi	58
BAB 6	PENGUJIAN	59
6.1	Sistematika Pengujian	59
6.2	Pengujian dan Pembahasan	59
6.2.1	Pengujian dan Analisa Banyaknya Generasi	59
6.2.2	Pengujian dan Analisa Ukuran Populasi	61
6.2.3	Pengujian dan Analisa Kombinasi P_c dan P_m	63
6.2.4	Pengujian dan Analisa Pengaruh Nilai P_m	65
6.2.5	Pengujian dan Analisa Pengaruh Nilai P_c	67
BAB 7	PENUTUP	71
7.1	Kesimpulan	71
7.2	Saran	71
DAFTAR	PUSTAKA	73

DAFTAR TABEL

Tabel 2. 1 Tabel kajian Pustaka	6
Tabel 2. 2 Tabel kajian Pustaka (lanjutan)	7
Tabel 2. 3 Populasi Awal	19
Tabel 2. 4 Himpunan <i>Offspring</i>	19
Tabel 2. 5 Populasi Baru Setelah Seleksi	19
Tabel 3. 1 Kandungan Nutrisi dan Harga Bahan Pakan	22
Tabel 3. 2 Kandungan Nutrisi dan Harga Bahan Pakan (Lanjutan)	23
Tabel 3. 3 Tabel Kebutuhan Zat Ayam Petelur	24
Tabel 3. 4 Himpunan Populasi Awal	27
Tabel 3. 5 Contoh representasi Kromosom	28
Tabel 3. 6 Tabel Populasi <i>Parent</i> dan <i>Offspring</i>	33
Tabel 3. 7 Komposisi Pakan Dalam Gram	34
Tabel 3. 8 Tabel Kandungan Nutrisi Bahan Pakan Terpilih	37
Tabel 3. 9 Hasil Perhitungan Penalti	38
Tabel 3. 10 Hasil Perhitungan <i>Fitness</i>	39
Tabel 3. 11 Tabel Populasi Baru	40
Tabel 4. 1 Rancangan Uji Coba Banyaknya Generasi	46
Tabel 4. 2 Rancangan Uji Coba Banyaknya Populasi	47
Tabel 4. 3 Rancangan Uji Coba Kombinasi P_c Dan P_m	47
Tabel 6. 1 Hasil Uji Coba Banyaknya Generasi	60
Tabel 6. 2 Hasil Uji Coba Ukuran Populasi	62
Tabel 6. 3 Hasil Uji Coba Kombinasi P_c dan P_m	64
Tabel 6. 4 Hasil Uji Coba Pengaruh nilai P_m	65
Tabel 6. 5 Hasil Uji Coba Pengaruh nilai P_m (Lanjutan)	66
Tabel 6. 6 Hasil Uji Coba Pengaruh nilai P_c	67
Tabel 6. 7 Tabel Kandungan Nutrisi Pakan DMC 882	69
Tabel 6. 8 Tabel Kandungan Nutrisi Komposisi Pakan Terbaik	69
Tabel 6. 9 Tabel Kebutuhan Nutrisi Ayam Fase Starter	70
Tabel 6. 10 Tabel Perbandingan Kandungan Nutrisi	70

DAFTAR GAMBAR

Gambar 2. 1 Struktur Umum Algoritme Genetika	15
Gambar 2. 2 Contoh <i>Single-Point Crossover</i>	17
Gambar 2. 3 Contoh <i>Reciprocal Exchange Mutation</i>	18
Gambar 3. 1 Tahapan Penelitian.....	20
Gambar 3. 2 Desain Arsitektur Aplikasi Optimasi Komposisi Pakan Ayam Petelur	24
Gambar 3. 3 <i>Flowchart</i> Optimasi Pakan Dengan Algoritme Genetika.....	26
Gambar 3. 4 <i>Flowchart</i> Membangkitkan Populasi Awal.....	28
Gambar 3. 5 <i>Flowchart</i> Proses <i>Crossover</i>	29
Gambar 3. 6 <i>Flowchart Single-Point Crossover</i>	30
Gambar 3. 7 <i>Single-Point Crossover</i>	30
Gambar 3. 8 <i>Flowchart</i> Proses Mutasi	31
Gambar 3. 9 <i>Flowchart Reciprocal Exchange Mutation</i>	32
Gambar 3. 10 <i>Reciprocal Exchange Mutation</i>	32
Gambar 3. 11 <i>Flowchart</i> Perhitungan Nilai <i>Fitness</i>	34
Gambar 3. 12 <i>Flowchart</i> Perhitungan Nilai <i>Fitness</i> (Lanjutan)	35
Gambar 3. 13 <i>Flowchart</i> Perhitungan Nilai Pinalti.....	36
Gambar 4. 1 Rancangan Antarmuka Halaman Awal.....	41
Gambar 4. 2 Rancangan Antarmuka <i>Tab</i> Proses Algoritme Genetika	42
Gambar 4. 3 Rancangan Antarmuka <i>Tab Crossover</i> Halaman Reproduksi	43
Gambar 4. 4 Rancangan Antarmuka Halaman Evaluasi.....	44
Gambar 4. 5 Rancangan Antarmuka Halaman Seleksi.....	45
Gambar 5. 1 Tampilan Halaman Awal.....	56
Gambar 5. 2 Halaman Inisialisasi	57
Gambar 5. 3 Halaman <i>Crossover</i> dan Mutasi.....	57
Gambar 5. 4 Halaman Evaluasi	58
Gambar 5. 5 Halaman Seleksi.....	58
Gambar 6. 1 Grafik Hasil Uji Coba Banyaknya Generasi	61
Gambar 6. 2 Grafik Hasil Uji Coba Ukuran Populasi.....	63
Gambar 6. 3 Grafik Hasil Uji Coba Kombinasi Pc dan Pm	64
Gambar 6. 4 Grafik Hasil Uji Coba Pengaruh Nilai Pm	66
Gambar 6. 5 Grafik Hasil Uji Coba Pengaruh Nilai Pc.....	68
Gambar 6. 6 Komposisi Pakan Terbaik Ayam Petelur	69

DAFTAR SOURCE CODE

<i>Source code</i> 5. 1 Membangkitkan Populasi Awal.....	49
<i>Source code</i> 5. 2 Proses <i>Crossover</i>	51
<i>Source code</i> 5. 3 Proses Mutasi.....	52
<i>Source code</i> 5. 4 Proses Perhitungan Penalti	54
<i>Source code</i> 5. 5 Proses Perhitungan <i>Fitness</i>	55
<i>Source code</i> 5. 6 Proses Seleksi	55



BAB 1 PENDAHULUAN

1.1 Latar Belakang

Peluang beternak ayam petelur dinilai cukup menjanjikan di Indonesia karena telur adalah bahan makanan yang sangat bermanfaat. Selain sebagai sumber protein hewani, telur juga sebagai campuran bahan kue, makanan ringan, dan banyak lainnya. Permintaan telur dalam negeri terus meningkat seiring dengan meningkatnya pola hidup manusia untuk memenuhi kebutuhan protein hewani. Berdasarkan data dari kementerian pertanian pada tahun 2017 terdapat peningkatan konsumsi telur ayam ras selama tahun 1987-2017 sebesar 3,57% pertahun dengan rata-rata konsumsi 6,63 kg/kap/th pada tahun 2017 (Kementerian Pertanian, 2017).

Dalam beternak ayam petelur maka ada 3 aspek yang perlu diperhatikan agar ayam mampu memproduksi telur secara maksimal. Aspek pertama yaitu bibit. Bibit ayam petelur harus dipilih dari bibit yang baik, bibit yang berkualitas, bibit yang tinggi produktivitasnya. Aspek kedua yang perlu diperhatikan dalam beternak ayam petelur yaitu manajemen. Manajemen pemeliharaan termasuk manajemen lingkungan dan kesehatan ayam petelur. Aspek ketiga yang perlu diperhatikan yaitu pakan. Pakan memegang peranan penting dalam beternak ayam petelur. Ayam petelur harus diberi pakan sesuai kebutuhan yang mencakup energi, air, mineral, lemak dan kandungan gizi lainnya sesuai rekomendasi, pakan tidak berbau, tidak berjamur, terbebas dari benda yang mengandung plastik, besi, kaca dan sejenisnya yang tidak bermanfaat bagi ayam petelur (Zulfikar, 2013).

Diantara bibit, manajemen pemeliharaan dan pakan ayam petelur, pakan ayam membutuhkan biaya terbesar dalam usaha beternak ayam petelur. Biaya pembelian pakan ternak ayam petelur dapat mencapai 60 – 70% dari seluruh biaya produksi (Abidin, 2004). Seringnya terjadi naik turun harga pakan ayam petelur membuat kondisi produksi yang tidak stabil. Khususnya, harga pakan ayam petelur yang semakin mahal tetapi hasil produksi tidak sebanding dengan biaya produksi yang dikeluarkan untuk pembelian pakan. Mahalnya harga pakan ayam petelur karena selama ini Indonesia masih mengimpor kebutuhan bahan baku pakan ternak ayam potensial seperti bungkil kedelai, tepung ikan, dan sebagian jagung (Murtidjo, 2006).

Untuk pemberian pakan ayam petelur, dapat diberikan pakan dengan beberapa kombinasi bahan pakan. Pemberian kombinasi bahan pakan sangat diperlukan agar kebutuhan nutrisi ayam petelur terpenuhi dengan baik. Tanpa kombinasi pakan, sulit bagi ayam petelur untuk memproduksi dan berprestasi dengan baik karena kebutuhan nutrisi ayam petelur nantinya akan kurang terpenuhi secara maksimal (Rasyaf, 2001).

Untuk menyelesaikan masalah harga pakan ayam yang mahal dan kombinasi pakan, maka hal tersebut dapat diatasi dengan optimasi komposisi

pakan ayam petelur secara tepat. Optimasi pakan dilakukan dengan mencampurkan beberapa jenis pakan baik yang dari hasil pabrik maupun bahan pakan alami sehingga dapat memenuhi nutrisi yang dibutuhkan oleh ayam petelur. Apabila nutrisi pakan ayam terpenuhi maka ayam petelur dapat memproduksi secara maksimal dengan biaya pembelian pakan yang minimal. Dengan adanya optimasi komposisi pakan ayam ini diharapkan peternak telur dapat menekan biaya pembelian pakan sehingga mampu memperoleh keuntungan produksi yang maksimal.

Salah satu metode untuk menyelesaikan masalah optimasi yaitu algoritme genetika. Algoritme genetika adalah tipe dari algoritme evolusi yang sering dipakai untuk menyelesaikan masalah optimasi dengan menghasilkan solusi yang mendekati optimum sehingga cocok diterapkan untuk optimasi komposisi pakan ternak ayam petelur. Proses dalam algoritme genetika diawali dengan inisialisasi, reproduksi, evaluasi kemudian seleksi untuk memilih individu dengan nilai *fitness* sejumlah himpunan populasi awal dari himpunan populasi dan *offspring* untuk dipertahankan hidup pada generasi selanjutnya (Mahmudy, 2013).

Penelitian ini merupakan perbaikan untuk penelitian yang sudah ada sebelumnya yang dilakukan oleh Ervina Marginingtyas mengenai optimasi pakan ayam petelur dengan algoritme genetika untuk memenuhi kebutuhan nutrisi ayam petelur dengan biaya yang minimum. Pada penelitian Ervina digunakan representasi *real code* dimana setiap kromosom memiliki panjang sebanyak data bahan pakan yang digunakan yaitu 40 bahan pakan, sedangkan pada penelitian penulis representasi kromosom menggunakan representasi permutasi dengan panjang kromosom sejumlah dengan banyaknya bahan pakan yang dipilih oleh pengguna dari 50 bahan pakan yang tersedia. Perbaikan yang lain yaitu pada pemilihan fase ayam petelur. Pada penelitian yang dilakukan oleh Ervina fase ayam yang digunakan yaitu fase *layer*, sedangkan pada penelitian penulis fase ayam dapat dipilih oleh pengguna yang meliputi fase *starter*, *grower*, dan *layer*. Hasil akhir dari penelitian Ervina berupa kombinasi bahan pakan terbaik dengan nutrisi terpenuhi dan biaya minimum (Marginingtyas, Mahmudy, 2015).

Penelitian lainnya yaitu penelitian yang dilakukan Rheza Raditya Andrianto mengenai optimasi komposisi pakan kuda dewasa menggunakan algoritme genetika. Perbedaan yang akan dilakukan pada penelitian ini yaitu pada penelitian yang dilakukan Rheza Raditya Andrianto representasi kromosom menggunakan *real code*, sedangkan penelitian penulis menggunakan representasi permutasi. Pada penelitian yang dilakukan Rheza metode *crossover* menggunakan metode *extended intermediate*, sedangkan pada penelitian penulis metode *crossover* menggunakan metode *single-point crossover*. Hasil akhir dari penelitian Rheza berupa rekomendasi komposisi pakan kuda dewasa yang memenuhi nutrisi dengan biaya yang lebih rendah dari biaya yang digunakan oleh peternak (Andrianto, Muflikhah, Rahayudi, 2017).

Berdasarkan hal tersebut maka penulis mengambil judul "*Optimasi komposisi Pakan ternak Ayam Petelur Menggunakan Algoritme Genetika*". Algoritme genetika banyak digunakan pada masalah praktis yang berfokus pada pencarian parameter-parameter optimal (Rismawan dan Kusumadewi, 2007). Algoritme genetika telah berhasil diterapkan pada berbagai masalah terutama dalam permasalahan komposisi makanan. Oleh karena itu, pada skripsi ini algoritme genetika digunakan untuk optimasi komposisi pakan ayam petelur sehingga didapatkan komposisi pakan yang optimal dengan harga yang minimal.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah, maka rumusan masalah yang dijadikan objek penelitian untuk skripsi ini, antara lain:

1. Bagaimana hasil komposisi pakan ternak ayam petelur terbaik dengan harga minimal yang dihasilkan menggunakan algoritme genetika
2. Bagaimana pengaruh perubahan parameter algoritme genetika terhadap hasil komposisi pakan ternak ayam petelur
3. Bagaimana perbandingan hasil optimasi pakan ternak ayam petelur menggunakan sistem dengan hasil manual

1.3 Tujuan

Tujuan dari pembuatan skripsi ini, antara lain:

1. Mendapatkan hasil komposisi pakan ayam petelur terbaik dengan harga minimum menggunakan algoritme genetika
2. Mengetahui cara menentukan parameter kromosom yang efisien untuk mengoptimasi komposisi pakan ternak ayam petelur
3. Mengetahui perbedaan hasil optimasi pakan ternak ayam petelur menggunakan sistem dengan hasil manual

1.4 Batasan Masalah

Berdasarkan latar belakang masalah, berikut beberapa batasan masalah yang diberikan untuk menghindari melebarnya masalah yang akan diselesaikan:

1. Penelitian ini digunakan untuk mencari optimasi komposisi pakan ternak ayam petelur yang memenuhi nutrisi dengan harga pakan yang minimal
2. Penelitian ini menggunakan metode algoritme genetika tanpa membandingkan dengan metode yang lainnya. Proses algoritme genetika dengan representasi kromosom menggunakan representasi permutasi. Proses *crossover* menggunakan *single-point crossover* sedangkan proses mutasi menggunakan *reciprocal exchange mutation*. Untuk proses seleksi menggunakan metode seleksi *elitism*
3. Bahan pakan yang digunakan dalam skripsi ini sebanyak 50 bahan pakan
4. Jumlah nutrisi yang digunakan dalam skripsi ini sebanyak 5 nutrisi

1.5 Manfaat

Dengan implementasi algoritme genetika untuk optimasi komposisi pakan ayam petelur ini diharapkan peternak ayam petelur dapat mengetahui komposisi pakan ayam petelur yang memenuhi kebutuhan nutrisi ayam dengan biaya yang minimal. Dengan biaya pakan yang minimal maka peternak ayam mampu memperoleh keuntungan produksi yang maksimal.

1.6 Sistematika Penulisan

Sistematika penyusunan laporan skripsi ini dibagi menjadi tujuh bab dengan masing-masing bab diuraikan sebagai berikut:

BAB I : Pendahuluan

Bab ini menjelaskan mengenai latar belakang masalah, perumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian dan sistematika penyusunan laporan.

BAB II : Landasan Kepustakaan

Bab ini menjelaskan dasar teori, konsep, model, metode yang berhubungan dengan optimasi komposisi pakan ternak ayam petelur dan algoritme genetika serta teori atau penelitian yang menggunakan metode algoritme genetika yang dapat memberikan solusi dalam permasalahan ini.

BAB III : Metodologi

Bab ini menjelaskan metode-metode yang digunakan dalam proses perancangan sistem optimasi komposisi pakan ayam petelur sehingga mampu memecahkan permasalahan mengenai optimasi komposisi pakan ternak ayam petelur dengan menggunakan algoritme genetika.

BAB IV : Perancangan

Bab ini menjelaskan perancangan dari aplikasi optimasi pakan ayam petelur menggunakan algoritme genetika. Perancangan meliputi perancangan database jika ada, perancangan antarmuka dan perancangan pengujian.

BAB V : Implementasi

Bab ini menjelaskan hasil dari implementasi algoritme genetika untuk optimasi komposisi pakan ternak ayam petelur dari perancangan optimasi pakan ternak ayam petelur dengan menggunakan algoritme genetika.

BAB VI : Pengujian

Bab ini menjelaskan analisis dan menguraikan hasil pengujian untuk menjawab pertanyaan atau masalah yang diteliti.

BAB VII : Penutup

Bab ini menjelaskan kesimpulan yang diperoleh dari implementasi algoritme genetika untuk optimasi komposisi pakan ayam petelur dan memberikan saran untuk mengembangkan penelitian ini.

BAB 2 LANDASAN KEPUSTAKAAN

2.1 Kajian Teori

Pada subbab kajian teori dijelaskan beberapa hasil penelitian yang relevan untuk mendukung penelitian ini. Beberapa penelitian tersebut memiliki kaitan dengan penelitian ini seperti ditunjukkan pada tabel 2.1.

Penelitian pertama sebagai rujukan utama penelitian ini yaitu penelitian yang sudah dilakukan oleh Ervina Marginingtyas mengenai optimasi pakan ayam petelur untuk mencari biaya minimum dengan algoritme genetika. Perbedaan pada penelitian tersebut yaitu penelitian tersebut hanya untuk ayam fase *layer*, Sedangkan pada penelitian penulis menggunakan fase ayam *starter*, *grower*, dan *layer*. Pada penelitian yang dilakukan oleh Ervina Marginingtyas menggunakan 40 bahan pakan dengan representasi *real code* dimana setiap kromosomnya memiliki panjang yang sama dengan jumlah data bahan pakan yang digunakan pada penelitian tersebut yaitu 40 bahan pakan (Sari, Mahmudy, 2014). Sedangkan penelitian penulis menggunakan representasi permutasi dengan panjang kromosom sejumlah dengan banyaknya bahan pakan yang dimasukkan oleh pengguna. Keunggulan dari penelitian penulis yaitu komposisi bahan pakan bisa dipakai untuk semua fase ayam petelur dari fase *starter*, *grower* dan *layer*. Representasi kromosom menggunakan representasi permutasi dimana panjang kromosom adalah banyak bahan pakan yang dipilih oleh pengguna. Hal ini lebih menguntungkan karena pengguna dapat memilih bahan pakan apa saja yang tersedia di lingkungan pengguna. Pada penelitian yang dilakukan oleh Ervina Marginingtyas tidak disebutkan hasil biaya optimum yang diperoleh dari hasil optimasi pakan serta tidak terdapat perbandingan harga pakan hasil optimasi dengan harga pakan dari peternak (Marginingtyas, Mahmudy, 2015).

Penelitian kedua yaitu penelitian yang dilakukan oleh Rheza Raditya Andrianto mengenai optimasi komposisi pakan kuda dewasa menggunakan algoritme genetika dewasa pada tahun 2017. Pada penelitian yang dilakukan Rheza Raditya Andrianto metode representasi kromosom menggunakan menggunakan metode representasi *real code*, metode *crossover* menggunakan metode *extended intermediate*, metode mutasi dengan *reciprocal exchange mutation*, dan metode seleksi menggunakan metode *elitism*. Berdasarkan hasil pengujian populasi yang dilakukan, diperoleh parameter optimal pada populasi 70 dengan rata-rata nilai *fitness* yaitu 0,17119007. Dari pengujian generasi didapatkan jumlah generasi optimal yaitu 250 dengan rata-rata nilai *fitness* 0,188259872. Sedangkan untuk pengujian kombinasi nilai *cr* dan *mr* didapatkan kombinasi *cr* 0,5 dan kombinasi *mr* 0,5 dengan rata-rata nilai *fitness* 0,188881. Dari hasil pengujian yang dilakukan dengan parameter terbaik menunjukkan bahwa sistem mampu merekomendasikan komposisi pakan kuda dengan biaya Rp.40316. Biaya yang didapatkan dari sistem lebih murah 43% dibandingkan dengan biaya yang digunakan peternak sekitar Rp.70000 (Andrianto, Muflikhah, Rahayudi, 2017).

Penelitian ketiga adalah penelitian yang dilakukan oleh Luh Kesuma Wardhani dkk mengenai algoritme genetika multi objek pada masalah optimasi komposisi pakan ikan air tawar pada tahun 2011. Dalam penelitian ini dibahas mengenai permasalahan dalam penentuan komposisi bahan pakan ikan tawar yang dinilai cukup penting karena tidak semua komposisi pakan untuk ikan air tawar mampu memberikan hasil terbaik untuk pertumbuhan ikan dan efisiensi biaya produksi. Dalam penelitian ini data yang digunakan adalah data pakan ikan patin siam dan ikan lele dumbo. Metode representasi kromosom menggunakan *integer encoding*, metode seleksi menggunakan *roulette wheel selection* dan metode *crossover* menggunakan metode *one-point crossover* (Wardhani, dkk, 2011).

Pada penelitian ini terdapat pengujian untuk mengukur tingkat keberhasilan metode *multi objective genetic algorithm* untuk menghasilkan pakan ikan lele dan ikan patin siam sebanyak 6 kilogram pada fase dewasa. Pengujian yang dilakukan dengan cara menguji parameter populasi, nilai p_c dan nilai p_m . Parameter generasi, gen, probabilitas *crossover* (P_c), dan probabilitas pilihan (P_e) yang digunakan pada setiap pengujian adalah sama yaitu jumlah generasi 5, panjang gen 5, dan probabilitas pilihan 0,03. Hasil pengujian untuk menghasilkan pakan sejumlah 6 kg untuk ikan lele dapat dicapai dengan parameter populasi 200, p_c 0,01, dan p_m 0,2. Dengan parameter tersebut didapatkan tingkat pemenuhan gizi mencapai 100% dan tingkat efisiensi biaya mencapai 46.5% untuk penentuan pakan ikan lele. Sedangkan untuk ikan patin siam dewasa diperoleh rata-rata tingkat keberhasilan pemenuhan gizi pakan mencapai 100% dan tingkat efisiensi biaya mencapai 42.75%. Kombinasi optimal untuk menghasilkan pakan sejumlah 6 kg untuk ikan patin siam dapat dicapai dengan parameter jumlah populasi 200, p_c 0,75, dan p_m 0,03 (Wardhani, dkk, 2011).

Tabel 2. 1 Tabel Kajian Pustaka

No.	Judul	Masukan	Metode	Hasil
1.	Penentuan Komposisi Pakan Ternak Untuk Memenuhi Kebutuhan Nutrisi Ayam Petelur Dengan Biaya Minimum Menggunakan Algoritme Genetika (Sari, Mahmudy, Dewi, 2015)	Ukuran populasi, jumlah generasi, <i>crossover rate</i> , <i>mutation rate</i> , berat ayam dan produksi telur	Algoritme genetika	Hasil pengujian diperoleh parameter terbaik untuk menghasilkan komposisi pakan yang optimal dengan harga yang minimal adalah populasi 160, generasi 1250, serta kombinasi cr 0,6 dan mr 0,4.

Tabel 2.2 Tabel kajian Pustaka (lanjutan)

No.	Judul	Masukan	Metode	Hasil
2.	Optimasi Komposisi Pakan Kuda Dewasa Menggunakan Algoritme genetika (Andrianto, Muflikhah, Rahayudi, 2017).	Fase, berat badan kuda, parameter algortime genetika	Algoritme genetika	Sistem mampu merekomendasikan komposisi pakan kuda dengan biaya Rp.40316. Biaya yang didapatkan dari sistem lebih murah 43% dibandingkan dengan biaya yang digunakan peternak sekitar Rp.70000
3.	Optimasi Komposisi Bahan pakan ikan Air Tawar Menggunakan Metode <i>Mutli-Objective Genetic Algorithm</i> (Wardhani, Safrizal, Chairi, 2011)	Jenis, umur, jenis kelamin ikan dan jumlah pakan yang dihasilkan, jumlah generasi maksimum, populasi, panjang kromosom, probabilitas <i>crossover</i> , probabilitas mutasi, dan seleksi <i>elitism</i>	<i>Mutli-objective genetic algorithm</i>	Algoritme genetika mampu diterapkan dengan rata-rata tingkat keberhasilan pemenuhan gizi mencapai 100% dan tingkat efisiensi biaya mencapai 46.5% untuk optimasi pakan ikan lele. Pada optimasi pakan ikan patin siam rata-rata tingkat keberhasilan pemenuhan gizi mencapai 100% dan tingkat efisiensi biaya mencapai 42.75%

2.2 Ayam Petelur

Ayam petelur merupakan ayam peliharaan (*Gallus gallus domesticus*). Ayam peliharaan biasanya dipelihara orang untuk dimanfaatkan untuk keperluan hidup pemeliharanya. Ayam peliharaan merupakan keturunan langsung dari subspecies ayam hutan yaitu ayam hutan merah (*Gallus gallus*) atau ayam bangkiwa (*bankiva fowl*). Adanya kawin silang antar ras ayam sehingga menghasilkan ratusan ras unggul atau galur murni dengan bermacam-macam fungsi diantaranya yang paling umum adalah ayam potong (untuk dipotong) dan ayam petelur (untuk diambil telurnya) (Menegristek, 2000).

2.2.1 Sejarah Ayam Petelur

Ayam petelur biasanya adalah ayam betina dewasa yang dipelihara secara khusus untuk diambil telurnya. Awal mula ayam unggas berasal dari ayam hutan dan yang ditangkap kemudian dipelihara sehingga dapat bertelur cukup banyak. Pada perkembangan selanjutnya, memelihara ayam hutan mulai diseleksi ketat oleh para pakar. Tujuan seleksi yaitu untuk memperoleh produksi yang banyak. Ayam yang dipelihara dengan tujuan untuk diambil dagingnya dikenal dengan sebutan ayam broiler, sedangkan ayam yang dipelihara dengan tujuan diambil telurnya dinamakan ayam petelur. Ayam petelur dibagi menjadi dua jenis yaitu ayam petelur putih dan ayam petelur cokelat. Proses persilangan dan seleksi itu tidak dilakukan secara instan tetapi memakan waktu lama hingga menghasilkan ayam petelur seperti yang ada sekarang. Pada setiap persilangan, sifat baik pada ayam petelur dipertahankan dan sifat jelek pada ayam petelur dibuang sehingga menghasilkan ayam petelur unggul (Rasyaf, 2001).

Ayam ras petelur *white leghorn* yang memiliki ciri badan kurus adalah ayam yang pertama masuk dan mulai banyak dternakkan. Sebelumnya masyarakat antipati terhadap daging ayam ras. Antipati ini kemudian mulai hilang dengan meluasnya peternakan ayam broiler pada akhir tahun 1990-an. Merebaknya peternakan ayam broiler diikuti dengan meluasnya peternakan ayam petelur dwiguna/ ayam petelur cokelat. Hal ini lah yang menyadarkan masyarakat bahwa ayam ras mempunyai kelebihan sebagai ayam petelur handal dan juga sebagai ayam pedaging yang enak (Menegristek, 2000).

2.2.2 Fase Ayam Petelur

Menurut Dr. M. Halim Natsir, S.Pt. MP (Halim, 2009) ada 4 fase ayam petelur yaitu:

2.2.2.1. Fase Starter

Periode *starter* adalah ayam yang berusia 0 hingga 8 minggu. Pada periode *starter* anak ayam akan mengalami pertumbuhan yang relatif cepat. Pada masa ini pula pemberian pakan yang tepat sangat berpengaruh untuk ayam *starter* karena dengan pemberian pakan yang tepat akan menentukan ketika ayam petelur memasuki fase selanjutnya (Rasyaf, 2001). Pertumbuhan ayam petelur fase *starter* dipengaruhi oleh seleksi ketat yang mencakup keaktifan bergerak, nafsu makan yang baik, pertumbuhan cepat, bobot badan sama, tingkat kematian rendah, kaki kuat dan mata cerah (Siregar dan Sabrani, 1986).

2.2.2.2. Fase Grower

Fase *grower* adalah fase pada saat ayam berumur 8 hingga 22 minggu. Fase ini sangat berhubungan dengan sistem reproduksi dan produksi ayam tersebut sehingga pada fase ini perlu adanya control pertumbuhan dan keseragaman. Ciri-ciri fisik ayam fase *grower* yaitu ukuran tubuh yang semakin

bertambah dan bulu-bulu yang semakin lengkap serta tampaknya kelamin sekunder pada tubuh ayam (Rasyaf, 2001).

Ciri lain fase *grower* yaitu berkembangnya sistem reproduksi pada ayam dan juga bertumbuhnya sistem produksi. Pada fase ini ayam petelur harus diperhatikan dengan baik terutama pada faktor bahan pakan dan cahaya. Karena apabila gagal memperhatikan faktor tersebut maka akan memberi peluang kegagalan pada masa produksi (Siregar dan Sabrani, 1986).

2.2.2.3. Fase Layer1

Periode *layer1* adalah ayam yang berumur 22 sampai 52 minggu. Pada periode *layer1* ayam petelur dalam masa yang paling menghasilkan banyak telur. Factor yang harus diperhatikan agar ayam petelur mampu menghasilkan telur yang maksimal yaitu faktor ransum, manajemen lingkungan dan cahaya.

2.2.2.4. Fase Layer2

Periode *layer2* adalah ayam yang berumur lebih dari 52 minggu. Pada periode ini produksi telur mengalami penurunan.

2.2.3 Jenis-jenis Ayam Petelur

Jenis ayam petelur menurut badan riset dan teknologi (warintek, 2004) dibagi menjadi dua tipe:

1. Tipe Ayam Petelur Ringan.

Tipe ayam petelur ringan disebut juga dengan ayam petelur putih. Ciri-ciri ayam ini adalah memiliki badan yang kurus/ramping serta mata yang bersinar, Bulu berwarna putih dan memiliki jengger merah. Ayam petelur ringan banyak dijual di Indonesia dengan nama yang bermacam-macam. Dapat dipastikan bahwa setiap pembibit ayam petelur di Indonesia memiliki dan menjual ayam jenis ini. Kelebihan ayam petelur ringan yaitu mampu bertelur lebih dari 260 telur pertahun karena memang ayam tipe ini dikhususkan untuk bertelur saja. Kekurangan ayam petelur ringan yaitu sangat sensitif terhadap cuaca panas dan keributan. Apabila ayam berada dalam kondisi ini maka produksi telur akan turun secara cepat.

2. Tipe Ayam Petelur Medium.

Disebut ayam petelur medium karena bobot ayam ini berada diantara ayam petelur ringan dan ayam broiler. Ayam ini juga disebut ayam dwiguna karena kemampuannya dalam bertelur cukup banyak dan juga mampu menghasilkan daging yang banyak pula. Sebutan lain untuk ayam ini yaitu ayam petelur coklat karena bulunya yang coklat dan hasil telurnya yang coklat pula. Kelebihan ayam ini yaitu hasil telurnya lebih digemari oleh masyarakat daripada telur putih walaupun secara gizi dan rasa sama. Selain itu masyarakat juga

meyukai ayam petelur medium karena daging ayam petelur medium lebih laku dijual sebagai ayam pedaging dengan rasa yang enak.

2.3 Nutrisi Pakan

Nutrisi pakan unggas yang baik tidak hanya sekedar ditinjau dari segi teknis saja, melainkan juga melibatkan segi ekonomis. Pakan unggas yang secara teknis memenuhi persyaratan nutrisi tetapi membutuhkan biaya yang mahal maka penyusunan pakan unggas tidak ekonomis. Begitu juga sebaliknya apabila biaya pakan murah tetapi tidak memenuhi persyaratan nutrisi berarti secara teknis tidak baik. Oleh karena itu, dalam penyusunan pakan unggas yang baik adalah menyusun dari segi teknis dan ekonomis yang memenuhi persyaratan (Rasyaf, 1992).

Agar produksi telur ayam sesuai dengan yang diharapkan, dibutuhkan kualitas pakan yang lebih baik dibandingkan dengan pakan ayam yang dipelihara untuk hobi. Nutrisi pakan yang dibutuhkan oleh ayam petelur antara lain energi, protein, lemak, mineral (Rasyaf, 1992).

2.3.1 Energi

Energi adalah kemampuan untuk melakukan pekerjaan. Sumber energi didalam tubuh berasal dari makanan yang dimakan. Didalam makanan yang dimakan terdapat energi kimia yang nantinya diubah menjadi bentuk energi lain didalam tubuh (Budianti, 2009). Energi yang didapat dari pakan yang dikonsumsi ternak akan digunakan untuk melakukan gerak hidup serta proses produksi lainnya. Saat makanan dikonsumsi ternak, bentuk energi diubah menjadi panas. Oleh sebab itu, hubungan antara energi dengan proses-proses hidup tubuh dinyatakan dalam satuan panas, yaitu kalori (Murtidjo, 2006).

Istilah energi yang umum digunakan dalam pakan ternak ayam yaitu energi metabolisme. Energi metabolisme adalah energi yang dapat dicerna setelah dikurangi dengan energi urine yang diekskresi. Tinggi atau rendahnya energi metabolisme dalam pakan ternak unggas mempengaruhi banyak sedikitnya ternak unggas mengkonsumsi pakan. Pakan yang memiliki energi tinggi akan lebih sedikit dikonsumsi ternak unggas. Sebaliknya untuk pakan dengan energi rendah akan lebih banyak dikonsumsi ternak unggas (Aksi, 2003).

Peranan energi dalam pakan ternak unggas umumnya dikonsumsi ternak untuk memenuhi kebutuhan energi. Baik untuk bekerja aktif dan bergerak bebas, atau dapat diubah menjadi jaringan-jaringan tubuh seperti bentuk otot-otot daging, bulu, produksi daging atau telur dan pemeliharaan tubuh. Apabila ada kelebihan energi, baru dimanfaatkan untuk pertumbuhan dan produksi (Murtidjo, 2006).

Ternak unggas tipe petelur perlu memperhitungkan kebutuhan energi pemeliharaan tubuh dan pertumbuhan usia menjelang bertelur karena

pemborosan energi akan tersimpan dalam bentuk lemak sehingga menghambat kemampuan produksi telur dan menghambat pertumbuhan (Murtidjo, 2006).

2.3.2 Protein

Protein berasal dari Bahasa Yunani *protos* yang artinya paling utama. Protein merupakan senyawa organik berbobot molekul tinggi yang merupakan polimer dari monomer-monomer asam amino yang dihubungkan satu sama lain dengan ikatan peptida. Kandungan protein mencakup karbon, nitrogen, oksigen, dan senyawa lain seperti fosfor dan sulfur. Pada makhluk hidup protein memiliki peran penting dalam struktur dan fungsi sel (PENS).

Molekul protein tersusun dari satuan-satuan dasar kimia yaitu asam amino. Asam amino didapatkan sebagai hasil akhir hidrolitik bila protein dimasak atau bila protein enzim-enzim tertentu bekerja pada protein tersebut. Ternak unggas membuat protein jaringan tubuhnya terutama dari asam-asam amino hasil pencernaan protein dan makanannya. Sebelum protein bahan makanan dapat diserap dan digunakan oleh unggas, zat tersebut harus dirombak ke dalam asam amino selama pencernaan (Rasyaf, 1992a).

Protein berperan penting bagi kehidupan ternak ayam petelur karena protein merupakan protoplasma aktif dalam sel hidup. Ayam petelur dengan konsumsi protein yang cukup akan mampu memperbaiki jaringan tubuhnya, menumbuhkan jaringan baru, metabolisme ke dalam zat-zat vital dalam fungsi tubuh, mengaktifkan enzim dan hormon tertentu sehingga tubuh dapat berfungsi dengan normal (Murtidjo, 2006).

Protein dibagi menjadi dua yaitu protein nabati dan protein hewani. Protein nabati berasal dari tumbuh-tumbuhan. Protein nabati umumnya memiliki kualitas protein rendah kecuali bungkil kedelai yang diketahui memiliki kualitas protein yang tinggi. Sedangkan untuk protein hewani berasal dari hewan yang diketahui memiliki kualitas protein tinggi, seperti pada tepung bulu unggas hidrolisa dan susu bubuk (Murtidjo, 2006).

Kekurangan konsumsi pakan yang mengandung protein bagi ayam petelur yaitu dapat menurunkan tingkat pertumbuhan ayam petelur. Apabila ternak ayam petelur sampai mengalami kekurangan protein yang parah akan menyebabkannya segera berhentinya pertumbuhan dan ternak akan kehilangan pertumbuhan sebesar 6-7% dari berat badan per hari (Abun, 2006).

Kelebihan protein juga tidak membawa manfaat yang lebih bagi ayam petelur. Kelebihan protein ditandai dengan penurunan pertumbuhan yang ringan, menurunnya timbunan lemak tubuh, naiknya tingkat asam urat dalam darah, meningkatnya *adrenocortocosteroid*, kelenjar adrenal membesar, dan kotoran ayam menjadi basah karena banyak konsumsi air minum (Abun, 2006).

2.3.3 Lemak

Lipid yang berarti lemak adalah senyawa organik yang terdapat di alam dan sukar larut dalam air, tetapi mudah larut dalam pelarut organik nonpolar, misalnya hidrokarbon atau eter. Lemak dikelompokkan berdasarkan kemiripan sifat fisiknya, sedangkan rumus kimia, fungsi, struktur, dan gugus fungsinya beraneka ragam. Senyawa yang tergolong *lipid* antara lain trigliserida (lemak dan minyak), fosfolipid dan steroid (Winarti, 2006).

Lemak merupakan senyawa heterogen yang dapat ditemukan dalam jaringan tanaman dan hewan. Lemak tersimpan dalam tubuh hewan, sedangkan minyak tersimpan dalam jaringan tanaman. Fungsi lemak yaitu sebagai sumber energi yang utama untuk metabolisme tubuh. Sumber lemak didalam tubuh diperoleh dari makanan dan produksi organ hati. Lemak disimpan didalam tubuh sebagai bentuk cadangan energi bagi makhluk hidup (Madja, 2007).

Perlunya lemak yang cukup didalam pakan ayam petelur adalah agar ayam memiliki cadangan energi selain karbohidrat. Karena energi metabolisme yang terkandung didalam lemak mencapai 2,25 kali lebih banyak dari karbohidrat. Didalam lemak terdapat asam lemak utama yang terdiri dari asam-asam palmitat, stearat, linoleat, dan oleat. Keempat asam lemak tersebut meliputi lebih dari 90% sebagian besar bahan pakan dan lemak karkas. Yang termasuk asam lemak jenuh yaitu palmitat dan stearat, Sedangkan linoleat dan oleat dinamakan asam lemak tidak jenuh (Murtidjo, 2006).

Dari keempat asam lemak tersebut (palmitat, stearat, oleat, linoleat) semuanya dapat disintesis oleh unggas, kecuali linoleat. Oleh sebab itu linoleat harus ada dalam ransum. Kekurangan asam linoleat dalam pakan ayam petelur menyebabkan suatu penyakit defisiensi dengan tanda-tanda seperti pertumbuhan anak ayam yang terganggu, produksi telur berkurang, telur kecil dan daya tetas rendah, ketahanan tubuh yang berkurang terhadap infeksi pernafasan (Insani, 2009).

Secara keseluruhan penggunaan lemak dalam ransum ayam petelur dimaksudkan agar ayam petelur mampu menghasilkan telur yang lebih besar dan daya produksi telur lebih tinggi. Penggunaan lemak dalam pakan ayam akan membuat pakan tidak mudah berdebu, membuat warna pakan menjadi lebih menarik, mempertinggi palatabilitas dan mencegah hilangnya zat-zat pakan akibat debu (Insani, 2009).

2.3.4 Mineral

Mineral adalah nutrisi yang sangat penting untuk kehidupan unggas dan makhluk hidup yang berada di air. Mineral dibedakan menjadi dua berdasarkan jumlah kebutuhan dan keberadaan dalam tubuh unggas yaitu makro mineral dan mikro mineral. Makro mineral terdiri dari kalsium, potasium, magnesium, sodium, klor, fosfor, dan sulfur. Mikro mineral terdiri dari mangan, besi, tembaga, kobalt, iodin, seng, selenium dan kromium (Rasyaf, 1992a).

Fungsi utama mineral dalam tubuh unggas menurut Murtidjo (Murtidjo, 2006) adalah sebagai berikut:

1. Zat peyusun untuk tulang dan gigi
2. Memelihara tekanan osmotik dan mengatur perubahan air dan larutan dalam tubuh unggas
3. Berfungsi untuk menyusun struktur jaringan lunak unggas
4. Berperan penting terhadap impuls syaraf dan otot
5. Memiliki peran penting untuk keseimbangan asam-basa tubuh serta mengatur pH darah dan cairan lainnya
6. Berfungsi sebagai komponen penting dari banyak enzim, vitamin, hormon, pigmen pernafasan
7. Sebagai kofaktor dalam metabolisme, katalis dan aktifator enzim

Kekurangan salah satu mineral dapat mengakibatkan demineralisasi pada tulang, pertumbuhan menurun, deformasi skeletal, pengapuran abnormal dari tulang rusuk dan sirip punggung, anoreksia, efisiensi pakan rendah, dan sebagainya. Pada ayam petelur kekurangan mineral dapat menyebabkan hilangnya fungsi berbagai enzim, meningkatkan nafsu makan, mengurangi produksi telur, kulit telur tipis, daya tetas telur menurun dan menghambat pertumbuhan tulang dan bulu pada ayam petelur (Japfa, 2004).

Hampir semua jaringan hewan dan tumbuhan merupakan sumber yang kaya mineral. Tetapi kandungan mineral pada jaringan tanaman lebih sedikit dibandingkan dengan kandungan mineral yang terdapat pada jaringan hewan. Mineral yang terdapat pada jaringan tanaman tidak dapat langsung digunakan karena pada jaringan tanaman terdapat asam phytic yang mengikat jaringan tanaman dengan senyawa organik lainnya. Agar dapat digunakan maka mineral perlu dihidrolisis dengan enzim atau pemanasan dan perendaman (Murtidjo, 2006).

2.4 Algoritme Genetika

Jhon Holland dari Universitas Michigan merupakan penemu algoritme genetika. Algoritme genetika terinspirasi dari teori evolusi yang kemudian diadopsi menjadi algoritme komputasi yang bisa digunakan untuk memecahkan suatu pencarian nilai dalam sebuah masalah optimasi. Algoritme ini didasarkan pada proses genetik yang ada dalam makhluk hidup, yaitu perkembangan generasi dalam sebuah populasi yang alami (PENS).

Algoritme genetika sudah diterapkan dalam berbagai masalah yang berfokus untuk mencari parameter optimal atau untuk mencari solusi yang optimal. Contoh penerapan algoritme genetika yaitu untuk merencanakan dan menjadwalkan produksi, kompresi citra, penjadwalan ruang kuliah, optimasi jadwal kuliah, dan untuk menyusun rute dan jadwal kunjungan wisata (Mahmudy, 2013).

Algoritme genetika memiliki 5 komponen dasar secara umum, yaitu (Michalewicz, 1996):

1. Representasi genetik dari beberapa solusi dari suatu permasalahan
2. Cara untuk menciptakan inisial populasi dari beberapa solusi
3. Evaluasi fungsi solusi dengan nilai *fitness* yang dimiliki masing masing individu
4. Beberapa operator genetika yang membagi beberapa anak selama proses reproduksi
5. Nilai untuk beberapa parameter dari algoritme genetika

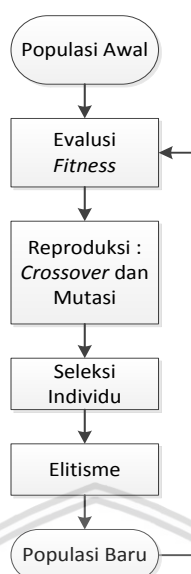
2.4.1 Kelebihan Algoritme Genetika

Kelebihan algoritme genetika sebagai metode optimasi (Mahmudy, 2013) adalah sebagai berikut:

1. Algoritme genetika dapat digunakan untuk masalah optimasi dengan ruang yang luas dan kompleks (Gen & Cheng, 1997)
2. Waktu komputasi untuk permasalahan yang kompleks dapat dikurangi dengan cara meletakkan individu pada sub-populasi yang diproses pada sejumlah komputer secara paralel (Defersha & Chen 2010; Qi, Burns & Harrison, 2000). Sub-populasi juga dapat diletakkan pada satu komputer untuk meningkatkan kualitas pencarian dan menjaga keragaman populasi (Mahmudy, 2013)
3. Untuk menyelesaikan masalah dengan banyak obyektif (Mahmudy & Rahman, 2011)
4. Mampu digunakan untuk menyelesaikan masalah dengan banyak variabel kontinyu, diskrit atau gabungan keduanya (Haupt & Haupt, 2004)
5. Mampu melakukan pencarian tanpa memperhatikan informasi derivative yang spesifik dari masalah yang diselesaikan (Gen & Cheng, 1997; Haupt & Haupt, 2004)
6. Dapat diimplementasikan pada berbagai macam data seperti data yang dibangkitkan secara numerik atau menggunakan fungsi analitis (Haupt & Haupt, 2004)
7. Cukup fleksibel sehingga dapat dihibridisasikan dengan algoritme lainnya (Gen & Cheng, 1997). Beberapa penelitian membuktikan bahwa *Hybrid Genetic Algorithms* (HGAs) sangat efektif untuk menghasilkan solusi yang lebih baik (Mahmudy, Marian & Luong 2013d, 2013c, 2014)
8. Algoritme genetika bersifat *ergodic*, sembarang solusi bisa diperoleh dari solusi yang lain dengan hanya beberapa langkah. Hal ini memungkinkan eksplorasi pada daerah pencarian yang sangat luas dilakukan dengan lebih cepat dan mudah (Marian, 2003)

2.4.2 Struktur Umum Algoritme Genetika

Struktur umum algoritma genetika untuk memecahkan suatu permasalahan menurut Michalewicz, 1996 ditunjukkan pada Gambar 2.1.



Gambar 2.1 Struktur Umum Algoritme Genetika

Solusi dari masalah direpresentasikan dalam bentuk string kromosom yang terdiri dari komponen genetik terkecil yaitu gen (PENS). Dari sejumlah solusi yang sudah direpresentasikan ke dalam bentuk kromosom ini dibangkitkan populasi awal secara acak. Sedangkan populasi selanjutnya merupakan hasil evolusi kromosom-kromosom melalui iterasi yang disebut dengan generasi.

Setelah populasi awal dibangkitkan maka masuk pada tahap evaluasi. Pada tahap evaluasi dihitung nilai *fitness* setiap kromosom. Nilai *fitness* dari suatu kromosom akan menunjukkan kualitas kromosom dalam populasi tersebut (Rismawan, Kusumadewi, 2007). Setelah nilai *fitness* dihitung sesuai dengan fungsi *fitnessnya*, maka dapat diketahui populasi mana yang tetap bertahan untuk generasi selanjutnya. Proses pemilihan individu dengan nilai *fitness* terbaik disebut dengan proses seleksi (Ariwibowo, 2008).

Proses selanjutnya yaitu reproduksi. Reproduksi dilakukan untuk menghasilkan keturunan (*offspring*) dari individu-individu yang ada di populasi (Mahmudy, 2013). Tahap reproduksi terdiri dari dua proses yaitu kawin silang (*crossover*) dan mutasi. Pada proses *crossover*, diambil beberapa kromosom dalam satu generasi kemudian disilangkan antar kromosom. Sehingga hasil persilangan tersebut masih mewarisi sifat kedua induknya dengan harapan kromosom yang disilangkan tersebut bersifat lebih baik dibandingkan generasi sebelumnya (Ariwibowo, 2008).

Proses mutasi adalah proses diubahnya salah satu atau lebih nilai gen didalam kromosom dalam suatu populasi. Nilai gen yang diubah akan diganti dengan suatu nilai gen lain yang dipilih secara acak. Akibat dari proses mutasi yaitu adanya kemungkinan sifat *offspring* yang dihasilkan tidak sama dengan sifat induknya (Ariwibowo, 2008).

Setelah populasi diproses sebanyak iterasi tertentu maka akan didapatkan individu terbaiknya. Hasil individu terbaik ini terdiri dari susunan kromosom yang mampu dikonversikan mendekati solusi terbaik yang mendekati optimum. Berdasarkan hal tersebut dapat disimpulkan bahwa algoritme genetika dapat menghasilkan solusi optimal dengan fungsi probabilitas (Michalewicz, 1996).

2.4.3 Komponen Utama Algoritme Genetika

Terdapat 6 komponen utama dalam algoritme genetika (PENS), yaitu:

2.4.3.1. Teknik Pengkodean

Teknik pengkodean adalah bagaimana kromosom dikodekan menjadi gen, dimana kromosom merupakan susunan dari banyak gen. Satu gen biasanya mewakili satu variabel. Bentuk representasi gen antara lain bentuk bilangan real, bit, daftar aturan, elemen program, elemen permutasi, atau representasi lainnya yang dapat diimplementasikan untuk operator genetika.

Contoh representasi kromosom antara lain:

- a. String bit : 100111, 110011, 100110, dan seterusnya
- b. Array bilangan real : 12.56, -34.67, 66.34 dan seterusnya
- c. Elemen permutasi : E3, E9, E8 dan seterusnya
- d. Daftar Aturan : Rule1, Rule2, Rule3 dan seterusnya
- e. Dan seterusnya

2.4.3.2. Membangkitkan Populasi Awal

Populasi awal biasanya didapatkan dengan cara membangkitkan sejumlah individu secara acak atau melalui alur tertentu. Banyak populasi yang akan dibangkitkan tergantung pada masalah yang akan diselesaikan dan jenis operator genetika yang akan diimplementasikan. Apabila ukuran populasi awal sudah ditentukan, maka populasi awal bisa dibangkitkan sebanyak ukuran yang ditentukan tersebut.

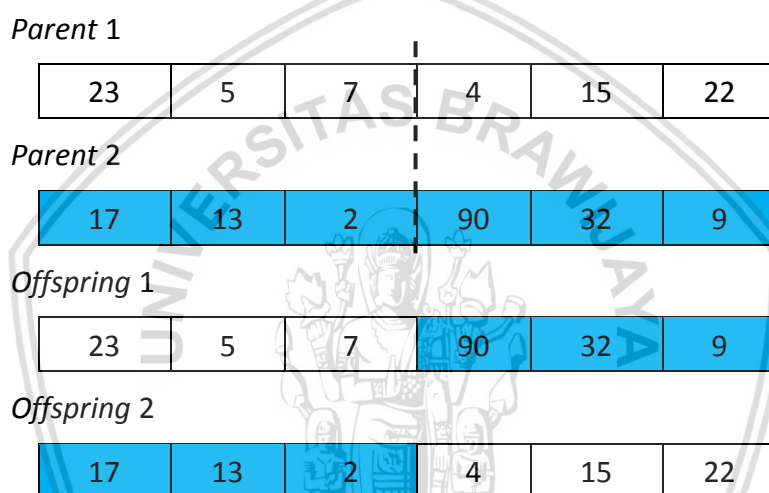
2.4.3.3. Kawin Silang (*Crossover*)

Kawin silang adalah operator dari algoritme genetika yang membutuhkan dua induk untuk membentuk suatu kromosom baru. Kawin silang menghasilkan titik baru dalam ruang pencarian yang siap diuji. Kawin silang tidak selalu dilakukan pada semua individu yang ada. Pemilihan individu induk dilakukan secara acak dengan nilai probabilitas *crossover* tertentu. Jika tidak dilakukan, maka nilai dari induk akan diturunkan kepada keturunan (*offspring*).

Dalam proses *crossover* jumlah populasi sangat mempengaruhi hasil dari proses *crossover*. Jumlah populasi yang sangat kecil akan berakibat buruk, yaitu akan menyebabkan suatu kromosom dengan gen-gen yang mengarah pada solusi akan sangat cepat menyebar ke kromosom-kromosom yang lain. Untuk

mengatasi masalah tersebut digunakan suatu aturan bahwa proses *crossover* hanya dapat dilakukan dengan membangkitkan bilangan *random* [0.1] dan nilai bangkitan *random* adalah kurang dari nilai probabilitas *crossover* yang telah ditentukan. Pada umumnya, nilai probabilitas *crossover* adalah mendekati 1 (Suyanto, 2010).

Pada penelitian ini digunakan metode *crossover* satu titik (*single-point crossover*). Metode ini secara acak memilih 1 titik potong pada 2 kromosom *parent* kemudian menukar bagian kanan dari 2 *parent* tersebut sehingga menghasilkan keturunan baru. Jika kromosom *parent* memiliki bentuk dan panjang kromosom yang sama maka operasi *crossover* dapat dilakukan dengan sekali tahap (Poli, Langdon, 1997). Contoh *single-point crossover* dengan titik potong pada titik ke 3 dapat di lihat pada Gambar 2.2.



Gambar 2.2 Contoh *Single-Point Crossover*

2.4.3.4. Mutasi

Mutasi adalah salah satu operator pada algoritme genetika. Peran operator ini yaitu untuk menggantikan gen yang hilang karena proses seleksi (Kusumadewi, 2003). Kromosom anak dimutasi dengan menambahkan nilai *random* yang sangat kecil (P_m) didefinisikan sebagai persentasi dari jumlah total gen pada populasi yang mengalami mutasi. Pengaruh probabilitas mutasi dalam proses mutasi yaitu untuk mengendalikan banyaknya gen baru yang akan dimunculkan untuk dievaluasi. Tetapi jika probabilitas mutasi terlalu kecil, banyak gen yang mungkin berguna tidak pernah dievaluasi. Sebaliknya jika probabilitas mutasi terlalu besar, maka akan terlalu banyak gangguan acak, sehingga anak akan kehilangan kemiripan dengan induknya (PENS).

Pada penelitian ini, metode mutasi yang digunakan adalah metode *reciprocal exchange mutation*. Cara kerja *reciprocal exchange mutation* adalah dengan memilih dua posisi gen secara acak kemudian menukar nilai gen tersebut (Mahmudy, 2013). Pada Gambar 2.3 terdapat contoh mutasi menggunakan metode *reciprocal exchange mutation*.

Parent

23	1	4	15	44
----	---	---	----	----

Offspring

23	1	44	15	4
----	---	----	----	---

Gambar 2.3 Contoh Reciprocal Exchange Mutation

2.4.3.5. Seleksi

Seleksi adalah memilih individu dengan nilai *fitness* terbaik sebanyak jumlah individu pada populasi awal untuk digunakan pada generasi berikutnya. Fungsi seleksi yaitu untuk memperoleh calon induk yang baik (PENS). Nilai *fitness* akan sangat berpengaruh pada proses seleksi karena semakin besar nilai *fitness* suatu individu maka akan semakin besar pula peluangnya untuk terpilih menjadi individu di generasi selanjutnya. Hal ini bertujuan agar populasi pada generasi selanjutnya terdiri dari individu yang lebih baik dari generasi sebelumnya. Metode seleksi yang banyak digunakan adalah *elitism*, *roulette wheel* dan *binary tournament* (Mahmudy, 2013).

Pada penelitian ini digunakan metode seleksi *elitism*. Cara kerja metode *elitism* yaitu dengan mengumpulkan semua individu ke dalam suatu populasi penampung yang terdiri dari himpunan individu *parent* dan himpunan individu *offspring*. Kemudian seleksi *elitism* akan memilih individu dengan nilai *fitness* terbaik sebanyak *popsiz* populasi awal. Keuntungan dari metode *elitism* yaitu dengan metode ini menjamin bahwa individu dengan nilai *fitness* terbaik akan lolos ke generasi selanjutnya. Langkah pertama untuk melakukan seleksi *elitism* yaitu dengan mencari nilai *fitness*. Pada nilai *fitness* tidak memiliki aturan baku. Pada penelitian ini semakin tinggi nilai *fitness* maka semakin optimal solusi yang diperoleh. Untuk menghitung nilai *fitness* dilihat terlebih dahulu apakah ada penalti atau tidak. Penalti yaitu adanya pelanggaran apabila kebutuhan nutrisi (Em, PK, SK, P dan Cl) kurang dari kebutuhan nutrisi yang disarankan. Persamaan menghitung penalti dapat dilihat pada Persamaan (2-1).

$$PN = \begin{cases} 0, & TotalNutrisi > KebNutrisi \\ KebNutrisi - TotalNutrisi, & TotalNutrisi < KebNutrisi \end{cases} \quad (2-1)$$

Angka pembilang pada rumus *fitness* ini menggunakan angka 1000 dikarenakan rata-rata harga pakan ayam petelur dalam ribuan. Perhitungan nilai *fitness* dapat dilihat pada Persamaan (2-2) sebagai berikut :

$$Fitness = \frac{1000}{(Harga + Penalti)} \quad (2-2)$$

Keterangan:

Harga = Total harga bahan makanan pada setiap kromosom

Penalti = Hasil penalti dari nutrisi (EM, PK, SK, P, Cl) bahan pakan

Langkah kedua apabila nilai fitness sudah didapatkan maka seleksi masuk ke tahap berikutnya (Mahmudy, 2013). Misalkan terdapat kumpulan individu dalam sebuah populasi dengan *popsiz* = 5 seperti di tunjuk pada Tabel 2.3.

Tabel 2.3 Populasi Awal

Individu	<i>Fitness</i>
P1	5
P2	4
P3	7
P4	10
P5	2

Terdapat juga himpunan *offspring* seperti yang ditunjuk pada Tabel 2.4.

Tabel 2.4 Himpunan *Offspring*

Individu	<i>Fitness</i>
C1	3
C2	8
C3	9

Maka akan di dapat himpunan individu yang lolos ke generasi selanjutnya ditunjukkan pada Tabel 2.5.

Tabel 2.5 Populasi Baru Setelah Seleksi

Individu Baru	Individu Asal	<i>Fitness</i>
P1	P4	10
P2	C3	9
P3	C2	8
P4	P3	7
P5	P5	5

2.4.4 Kondisi Berhenti (*Termination Condition*)

Iterasi pada algoritme genetika akan diulang terus sampai kondisi berhenti tercapai. Iterasi akan berhenti dengan beberapa kondisi seperti berikut:

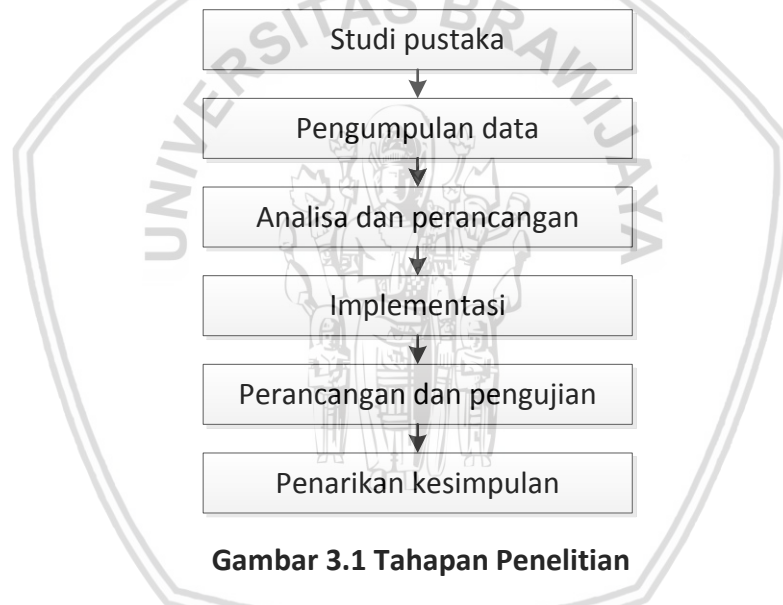
1. Iterasi berhenti sampai generasi *n*. Nilai *n* telah ditentukan sebelumnya berdasarkan beberapa pengujian. Nilai *n* akan semakin besar apabila ukuran dan kompleksitas masalah juga semakin tinggi (Yogeswaran, Ponnambalam & Tiwari, 2009)
2. Tidak dijumpai solusi yang lebih baik setelah generasi *n* (Mahmudy, Marian & Luong, 2012a). Pada kondisi ini algoritme genetika kesulitan untuk mendapatkan solusi yang lebih baik sedangkan penambahan iterasi hanya akan membuang waktu
3. Iterasi berhenti setelah waktu yang ditentukan terpenuhi. Hal ini biasa dipakai apabila kita ingin membandingkan performa dari beberapa algoritme untuk memecahkan suatu masalah (Mahmudy, 2013)

BAB 3 METODOLOGI

Dalam bab metodologi ini akan dibahas mengenai langkah-langkah pembuatan sistem optimasi komposisi pakan ayam petelur menggunakan algoritme genetika. Langkah-langkah dalam pembuatan sistem optimasi pakan ayam petelur menggunakan algoritme genetika adalah tahapan penelitian, metode pengumpulan data, analisa kebutuhan dan tahap-tahap penyelesaian masalah yang meliputi representasi kromosom, perhitungan nilai *fitness*, inisialisasi populasi awal, reproduksi dan seleksi.

3.1 Tahapan Penelitian

Tahapan penelitian untuk optimasi komposisi pakan ayam petelur menggunakan algoritme genetika terdiri dari studi literatur, pengumpulan data, analisa dan perancangan, implementasi dan pengujian. Adapun tahapan penelitian ditunjukkan pada Gambar 3.1.



Gambar 3.1 Tahapan Penelitian

Berdasarkan Gambar 3.1 tahap-tahap dalam penelitian berikut ini adalah sebagai berikut:

1. Studi pustaka bertujuan untuk memperoleh dasar teori dari berbagai sumber sebagai acuan untuk mendukung penulisan skripsi, seperti pengertian algoritme genetika, optimasi pakan komposisi pakan ayam petelur, *crossover*, mutasi, seleksi dan sebagainya
2. Pengumpulan data, bertujuan untuk memperoleh data dari pakar yang akan digunakan dalam penelitian yang dalam penelitian ini adalah data pakan ayam petelur
3. Melakukan analisa dan perancangan sistem sesuai dengan aplikasi yang dibuat
4. Mengimplementasi hasil analisa dan perancangan yang telah dibuat
5. Merancang pengujian dan analisis terhadap hasil implementasi yang dibuat

3.2 Studi Pustaka

Untuk mempelajari dasar teori yang berkaitan dengan optimasi komposisi pakan ayam petelur maka dibutuhkan studi literatur. Studi literatur dapat diperoleh dari bermacam-macam sumber diantaranya buku, e-book, internet, wawancara, jurnal, penelitian sebelumnya, dan sumber lainnya yang dapat dipertanggung jawabkan. Teori yang di pelajari terkait dengan penelitian ini adalah komposisi pakan ayam petelur, algoritme genetika, *crossover*, mutasi dan seleksi.

3.3 Pengumpulan Data

Pengumpulan data dilakukan untuk mendapatkan data dan informasi tentang optimasi pakan ayam petelur. Data yang diperoleh dari pakar dilengkapi dengan buku, laporan penelitian dan pengalaman pakar. Ada beberapa metode dalam pengumpulan data diantaranya yaitu dengan cara studi pustaka dan wawancara. Data yang digunakan dalam penelitian ini adalah data Kebutuhan nutrisi dan kandungan nutrisi pakan ayam petelur dari Dr. M. Halim Natsir, S.Pt. MP. Dosen peternakan bagian nutrisi dan makanan ternak. Bentuk data pakan dan nutrisi yang diberikan berupa data excel dengan jumlah data pakan sebanyak 100 data dan data nutrisi sebanyak 24 kandungan nutrisi. Dari data pakan yang diperoleh terpilih 50 bahan pakan yang memiliki kandungan nutrisi utama EM, Pk, SK, P, Cl yang baik untuk pertumbuhan ayam petelur dan mudah didapatkan oleh peternak. Harga pakan ayam petelur diperoleh berdasarkan survey pada tahun 2015 di kota malang.

3.4 Analisa dan Perancangan

Analisis kebutuhan bertujuan untuk menganalisis dan mendapatkan semua kebutuhan yang diperlukan dalam pengimplementasian algoritme genetika dalam optimasi komposisi pakan ayam petelur. Analisa kebutuhan pada aplikasi optimasi komposisi pakan ayam petelur menggunakan algoritme genetika meliputi analisa kebutuhan fungsional dan analisa kebutuhan data.

3.4.1 Analisa Kebutuhan

Dalam membuat suatu sistem maka perlu adanya analisa kebutuhan fungsional agar sistem optimasi komposisi pakan ayam petelur ini memiliki fungsi sesuai dengan kebutuhannya. Berikut adalah analisa kebutuhan fungsional dalam pembuatan aplikasi implementasi metode algoritme genetika untuk optimasi komposisi pakan ternak ayam petelur.

3.4.1.1 Analisa kebutuhan fungsional

- Aplikasi dapat melakukan masukan berupa fase ayam petelur, jumlah iterasi, jumlah populasi, nilai *cr*, nilai *mr*
- Aplikasi dapat memilih banyak bahan pakan dan jenis bahan pakan apa saja yang diinginkan oleh pengguna
- Aplikasi dapat menampilkan populasi awal, hasil *crossover*, hasil mutasi, dan hasil seleksi

- d. Aplikasi dapat menampilkan individu terbaik dari hasil perhitungan algoritme genetika
- e. Aplikasi dapat menampilkan hasil komposisi pakan ayam petelur yang optimal dengan harga yang minimal

3.4.1.2 Analisa data yang dibutuhkan meliputi:

- a. Data kandungan nutrisi pakan ayam petelur
- b. Data kebutuhan nutrisi pakan ayam petelur
- c. Harga bahan pakan ayam petelur

3.4.2 Deskripsi Permasalahan

Dalam beternak ayam petelur pakan merupakan faktor penting yang harus diperhatikan supaya ayam petelur dapat memproduksi telur secara maksimal. Biaya ransum untuk pakan ayam sendiri dapat mencapai 60% - 70% dari total biaya produksi secara keseluruhan. Seringnya terdapat fluktuasi harga produksi peternakan unggas akan menciptakan kondisi yang tidak stabil. Khususnya, harga pembelian pakan ayam petelur yang semakin mahal, di lain pihak harga hasil produksi ayam petelur tidak sebanding dengan biaya produksi yang dikeluarkan untuk pembelian pakan ayam petelur. Untuk itu perlu dilakukan optimasi komposisi pakan ayam petelur agar didapatkan solusi komposisi pakan yang memenuhi kebutuhan nutrisi secara maksimal tetapi dengan biaya yang minimal. Untuk mempermudah proses tersebut diperlukan adanya penerapan suatu sistem komputasi cerdas.

Pada penelitian ini digunakan 50 bahan pakan untuk ayam petelur. Pada Tabel 3.1 dan Tabel 3.2 terdapat daftar bahan pakan ayam petelur beserta kandungan gizi dan harga bahan makanan yang digunakan pada penelitian ini.

Tabel 3. 1 Kandungan Nutrisi dan Harga Bahan Pakan

NO.	BAHAN PAKAN	HARGA	EM	PK	SK	P	CI
		Rp/gr	Kkal/kg	%	%	%	%
1	Bekatul	10	2800	10	3	0.16	0.07
2	Beras pecah kulit	21	2660	8	9	0.04	0.09
3	Beras putih	12	3100	7.5	0.4	0.01	0.06
4	Dedak halus	9,5	2500	12	12	0.21	0.07
5	Dedak jagung	7,8	2950	10.6	5	0.15	0.07
6	Gandum	95	2980	10.7	2.1	0	0
7	Gula	13	3350	0	0	0	0
8	Jagung kuning	13	3370	9.00	4.76	0.1	0.04
9	Menir	14	3390	8.9	3	0.4	0
10	Pollard	6	1300	15	10	0.32	0.09
11	Sorghum	4,5	3250	10	2	0.1	0.08
12	Tepung gaplek	3,5	2900	1.5	0.9	0.09	0.07
13	Tetes (bit)	2	1980	6.5	0	0.2	0.7
14	Tetes (tebu)	2,7	1960	3	0	0.1	0.5
15	Whey	120	1910	13	0	0.8	0.7

Tabel 3.2 Kandungan Nutrisi dan Harga Bahan Pakan (Lanjutan)

NO.	BAHAN PAKAN	HARGA	EM	PK	SK	P	CI
		Rp/gr	Kkal/Kg	%	%	%	%
16	Bungkil biji kapas	2	2100	41	12	0.33	0.05
17	Bungkil biji karet	3	2159	24.2	9.8	0	0
18	Bungkil kedele	11,8	2200	44	6	0.65	0.03
19	Bungkil kelapa	4	2100	20	15	0.57	0.03
20	Bungkil wijen	6	1910	45	5	0.3	0.06
21	Bungkil.b.matahari	6	1760	31	21	0.3	0.17
22	Bungkil.kc. Tanah	3	2200	42	17	0.2	0.03
23	Butermilk kering	180	2730	32	0.4	0.9	0.7
24	Foka	40	2700	14	10.1	1	0.07
25	Hidrolisis i. Rumen	35	2000	16.2	25.4	0.55	0
26	Kacang buncis	10	2330	23.5	4.5	0.2	0.04
27	Kacang hijau	23	2220	21.30	0.92	0	0
28	Kacang kapri	25	2200	22	6	0.1	0.06
29	Kacang kedele	13	3510	38	5	0.25	0.03
30	Konsentrat broiler	8	2800	41	5	1.5	0
31	Konsentrat layer	10	2500	32	6	0	0
32	Mbm	15	2100	50	2.8	5.1	0.69
33	Ragi bir	78	1850	35	3	0.5	0.12
34	Ragi torula	49	1850	48	2	0.5	0.02
35	Susu skim	90	2510	33	0.2	1	0.9
36	T ikan(ancovetta)	18	2830	65	1	2.6	0.3
37	T. Daun alfalfa	45	1630	20	22	0.27	0.38
38	T.ikan (herring))	18	2640	72	1	1.5	1
39	T.ikan (menhaden)	18	2100	50	1	2.8	1.2
40	Tepung bekicot	50	4906	61	4.5	0	0
41	Tepung bulu ayam	6	2310	85	1.5	0.32	0
42	Tepung daging	7	2957	57	0	0	0
43	Tepung darah	7	2750	85	1	0.32	0.27
44	Tepung lamtoro	20	828	18.9	16.3	0	0
45	Dicalcium phosphat	8	0	0	0	0.1	0
46	Garam	3	0	0	0	0	32
47	Tepung tulang	6	818	12	2.3	13.5	0
48	Minyak ikan	17	8450	0	0	0	0
49	Pap	5	950	66.67	1.23	0	0
50	Komak	50	2200	21.04	1.00	0.1	0.06

Untuk kebutuhan zat ayam petelur ditunjukkan pada Tabel 3.3 berikut:

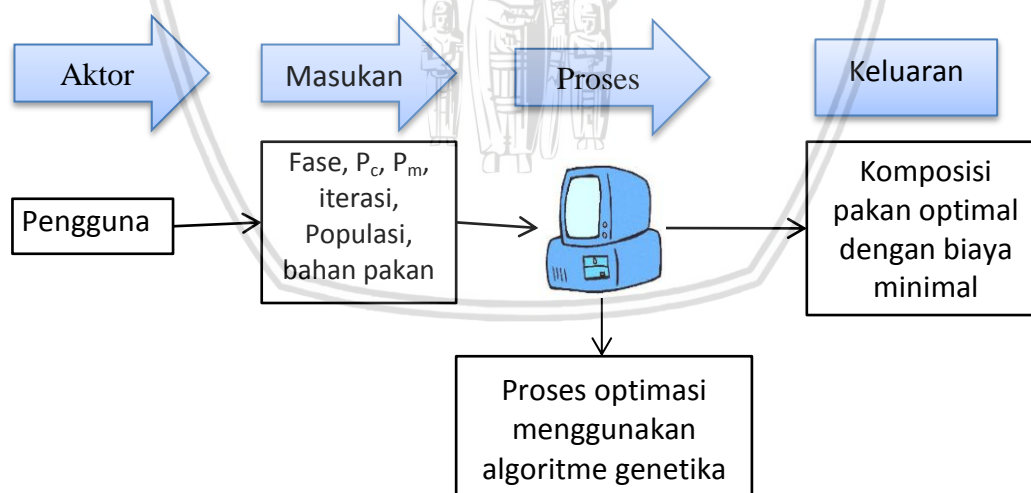
Tabel 3.3 Tabel Kebutuhan Zat Ayam Petelur

No.	Jenis Unggas	Periode / umur	Kebutuhan Zat Makanan				
			EM	PK	SK	P	CI
			Kkal/kg	%	%	%	%
1	Ayam Petelur	Starter (0-8mg)	2800	19	4-5	0.6-0.8	0.15
2	Ayam Petelur	Grower (8-22mg)	2600	16	5-6	0.6-0.8	0.12
3	Ayam Petelur	Layer1 (22-52mg)	2650	17	4-6	0.7-0.9	0.15
4	Ayam Petelur	Layer2 (>52mg)	2650	15.5	4-6	0.7-0.9	0.12

3.4.3 Deskripsi Umum Sistem

Secara umum aplikasi yang dibangun merupakan implementasi dari algoritme genetika untuk optimasi komposisi pakan ayam petelur. Untuk perhitungan pada algoritme genetika dalam optimasi komposisi pakan ayam petelur menggunakan *single-point crossover*, *reciprocal exchange mutation* dan *elitism selection*. Data jumlah pakan yang digunakan sebanyak 50 bahan pakan.

Masukan dari sistem ini adalah jenis dan fase ayam petelur, probabilitas *crossover*(p_c), probabilitas mutasi (p_m), *popsi*, iterasi dan bahan pakan yang diinginkan oleh pengguna. Sedangkan keluaran yang dihasilkan berupa solusi komposisi pakan ternak ayam petelur dengan komposisi yang tepat yaitu terpenuhinya kebutuhan nutrisi harian ayam petelur namun dengan biaya pembelian pakan yang minimal. Desain arsitektur aplikasi dapat dilihat pada Gambar 3.2.



Gambar 3.2 Desain Arsitektur Aplikasi Optimasi Komposisi Pakan Ayam Petelur

Pada Gambar 3.2 ditunjukkan urutan kerja aplikasi. Pertama pengguna akan memasukkan data fase, p_c , p_m , iterasi, populasi dan bahan pakan yang diinginkan sebagai masukan awal. Proses selanjutnya yaitu melakukan optimasi menggunakan algoritme genetika dengan membangkitkan populasi awal secara random dengan *popsi* sesuai masukan. Kemudian dilakukan *crossover* menggunakan *single-point crossover*. Kemudian dilakukan mutasi menggunakan

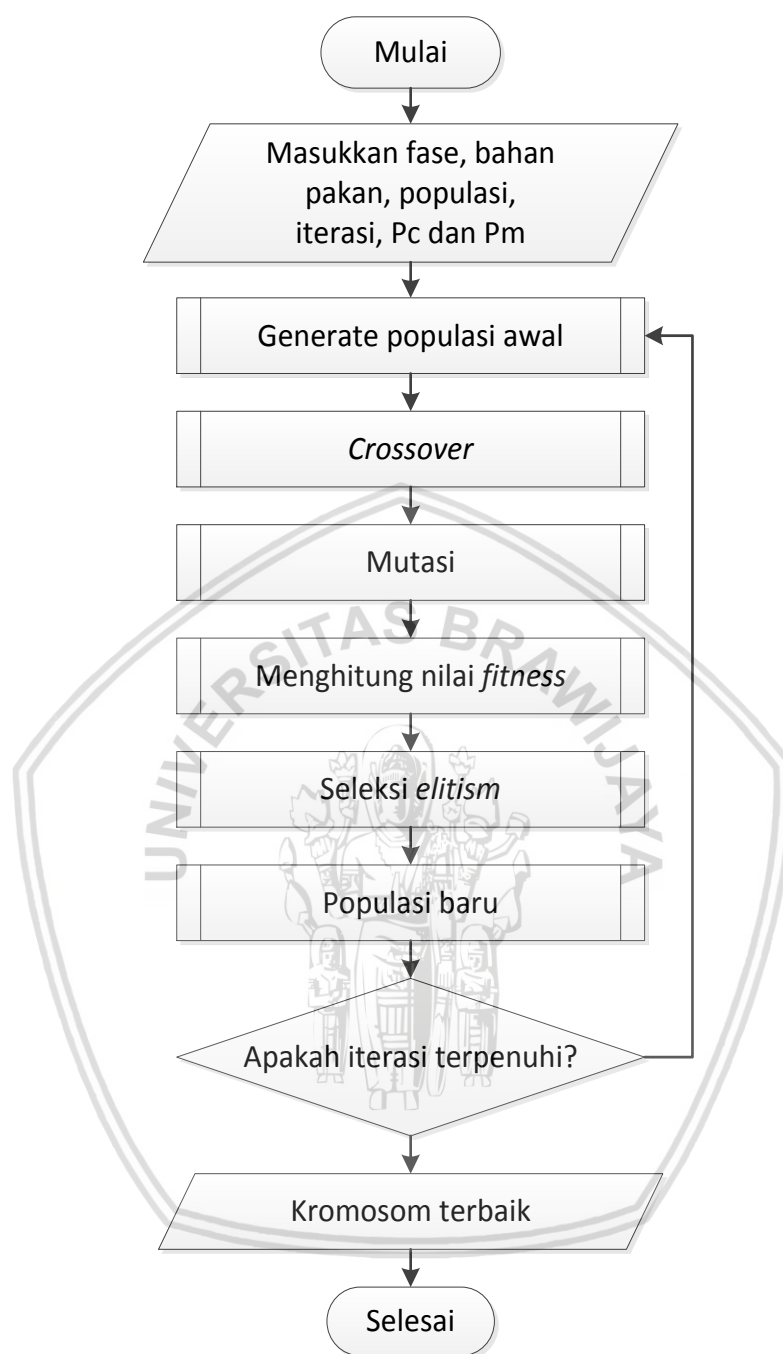
reciprocal exchange mutation. Setelah melakukan *crossover* dan mutasi, dihitung nilai *fitness* dari populasi awal dan populasi hasil *crossover* dan mutasi. Jika nilai *fitness* telah dihitung maka dilakukan seleksi individu terbaik menggunakan *elitism selection*. Dari hasil seleksi didapatkan populasi baru. Kemudian proses akan berulang sesuai dengan jumlah iterasi yang dimasukkan pengguna. Hasil akhir dari aplikasi ini berupa solusi komposisi pakan ternak ayam petelur dengan komposisi yang tepat yaitu terpenuhinya kebutuhan nutrisi harian ayam petelur namun dengan biaya pembelian pakan yang minimal.

3.4.4 Proses Perhitungan

Pada subbab proses perhitungan akan dijelaskan proses perhitungan optimasi dengan menggunakan algoritme genetika. Langkah-langkah perhitungan optimasi dengan menggunakan algoritme genetika adalah sebagai berikut:

1. Inisialisasi parameter awal, yaitu:
 - a. Parameter optimasi komposisi pakan adalah memasukkan fase ayam dan bahan pakan yang digunakan
 - b. Parameter algoritme genetika yaitu ukuran populasi, banyak generasi, probabilitas *crossover*, probabilitas mutasi
2. *Generate* populasi awal sebanyak ukuran populasi yang telah ditentukan.
3. Membuat populasi baru dengan langkah sebagai berikut:
 - a. Melakukan proses *crossover* pada induk yang terpilih secara acak berdasarkan p_c yang telah ditentukan
 - b. Melakukan Melakukan proses mutasi pada induk yang terpilih secara acak berdasarkan p_m yang telah ditentukan
 - c. Menghitung nilai *fitness* untuk masing masing kromosom pada nilai *offspring*
 - d. Menggabungkan populasi awal dengan himpunan *offspring* untuk diseleksi menjadi populasi baru untuk generasi selanjutnya
 - e. Seleksi menggunakan metode *elitism* untuk menentukan individu yang lolos untuk generasi selanjutnya
4. Kondisi perulangan akan berhenti apabila telah terjadi perulangan sebanyak jumlah generasi yang telah dimasukkan oleh pengguna.

Flowchart proses perhitungan algoritme genetika dapat dilihat pada Gambar 3.3.



Gambar 3.3 Flowchart Optimasi Pakan Dengan Algoritme Genetika

Misalkan pada penelitian ini terdapat masukan sebagai parameter awal sebagai berikut:

- Fase ayam = fase *starter*
- Kebutuhan pakan = 50 gr
- Ukuran populasi = 10
- Banyak generasi = 1

- Probabilitas *crossover*=0.2
- Probabilitas mutasi=0.1
- Bahan pakan yang terpilih yaitu bekatul, beras pecah kulit, beras putih, dedak halus dan dedak putih

Setelah diketahui kebutuhan nutrisi dalam satuan kkal dan persen maka dirubah ke dalam satuan gr dari kemampuan konsumsi ayam per harinya. Misalkan pada fase starter kemampuan konsumsi ayam 50 gram yaitu :

- $EM = 2800 \times 50 = 140000 \text{ gr}$
- $PK = \frac{19}{100} \times 50 = 9.5 \text{ gr}$
- $SK = \frac{5}{100} \times 50 = 2.5 \text{ gr}$
- $P = \frac{0.8}{100} \times 50 = 0.4 \text{ gr}$
- $CI = \frac{0.15}{100} \times 50 = 0.075 \text{ gr}$

Setelah itu langkah-langkah yang dilakukan dalam menerapkan metode algoritme genetika yaitu:

1. Membangkitkan populasi awal sebanyak jumlah populasi yang telah dimasukkan. Populasi awal dibangkitkan sesuai dengan jumlah populasi yang dimasukkan oleh pengguna dan panjang kromosom sesuai dengan banyak bahan pakan yang dimasukkan pengguna. Dalam kasus ini populasi yang dibangkitkan sebanyak 10 dengan panjang kromosom 5. Representasi nilai kromosom dilakukan dengan membangkitkan nilai integer pada interval antara 0-10 secara *random*. Nilai integer 0-10 mewakili bobot bahan pakan. Contoh populasi awal yang telah dibangkitkan dapat dilihat pada Tabel 3.4.

Tabel 3.4 Himpunan Populasi Awal

P(n)	K1	K2	K3	K4	K5	Total
P1	1	7	10	1	10	29
P2	0	6	3	10	8	27
P3	8	7	8	1	6	30
P4	3	9	2	1	1	16
P5	4	7	10	7	7	35
P6	4	10	10	1	1	26
P7	0	0	8	5	6	19
P8	0	9	3	2	0	14
P9	6	10	2	9	0	27
P10	4	5	1	5	6	21

Keterangan :

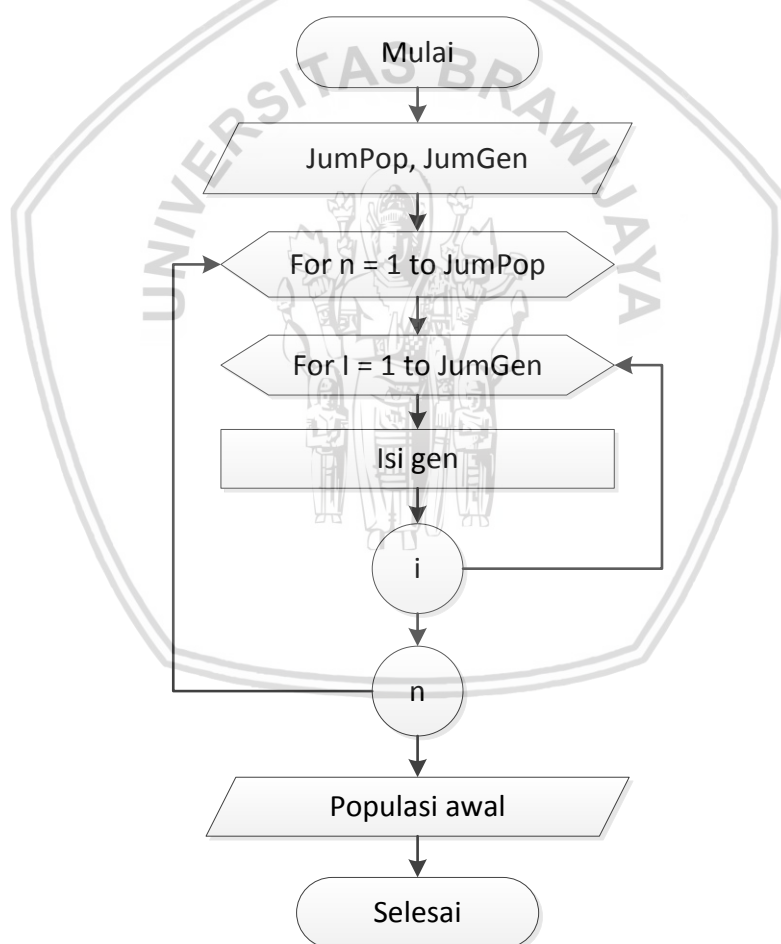
P(n) = *Parent*

K1,K2,K3,K4,K5 = Bahan makanan masukan ke-1 hingga masukan ke-5.
Contoh representasi permutasi kromosom pertama ditunjukkan pada Tabel 3.5 berikut:

Tabel 3.5 Contoh representasi Kromosom

P(n)	K1	K2	K3	K4	K5	Total
P1	1	7	10	1	10	29

Gen pertama pada representasi kromosom tersebut adalah 1 yang menunjukkan bobot pada bahan makanan terpilih 1 yang berdasarkan masukan pengguna yaitu bekatul. *Flowchart* untuk membangkitkan populasi awal ditunjuk pada Gambar 3.4.



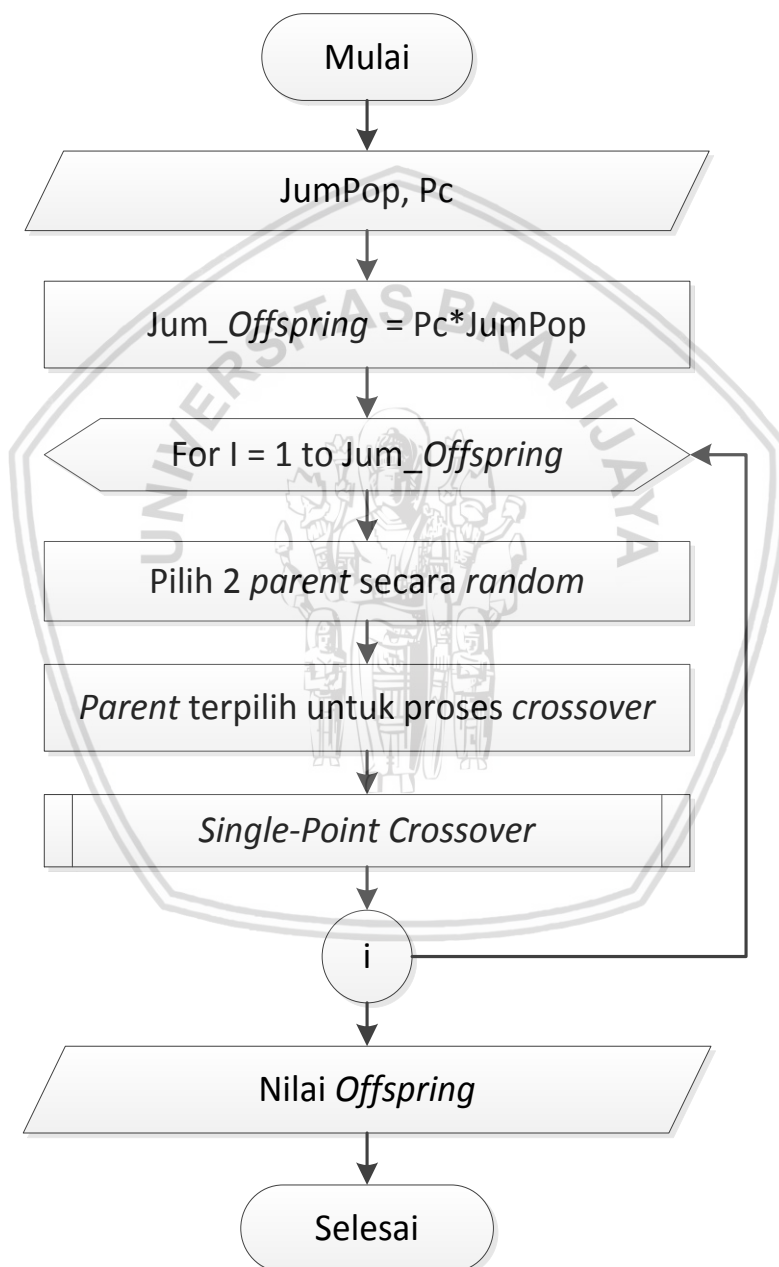
Gambar 3.4 Flowchart Membangkitkan Populasi Awal

- Langkah selanjutnya yaitu proses *crossover*. Pada penelitian ini di gunakan metode *crossover single-point crossover* yaitu akan dilakukan penukaran sebagian informasi dari 2 gen kromosom induk (*parent*) yang

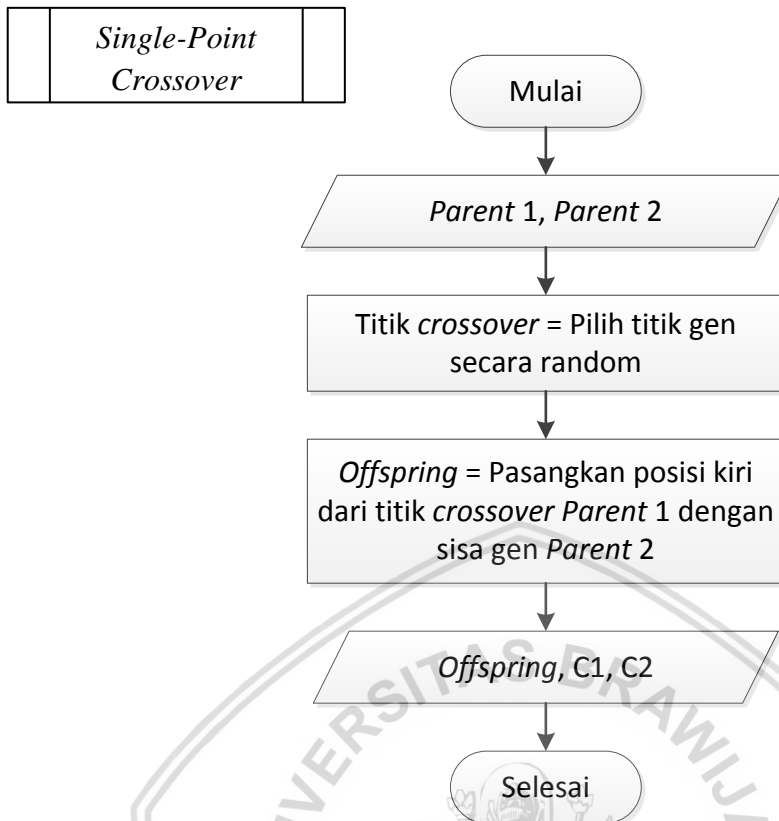
dipilih secara *random* dan akan menghasilkan *offspring* dengan 1 titik potong. Nilai *crossover rate* yang dimasukkan pengguna yaitu 0.2.

$$\begin{aligned}\text{Crossover} &= p_c * \text{popsize} \\ &= 0,2 * 10 \\ &= 2\end{aligned}$$

Sehingga, *Offspring* yang akan dihasilkan yaitu 2 dari 2 induk yang dipilih secara acak. *Flowchart crossover* dapat dilihat pada Gambar 3.5 dan 3.6.

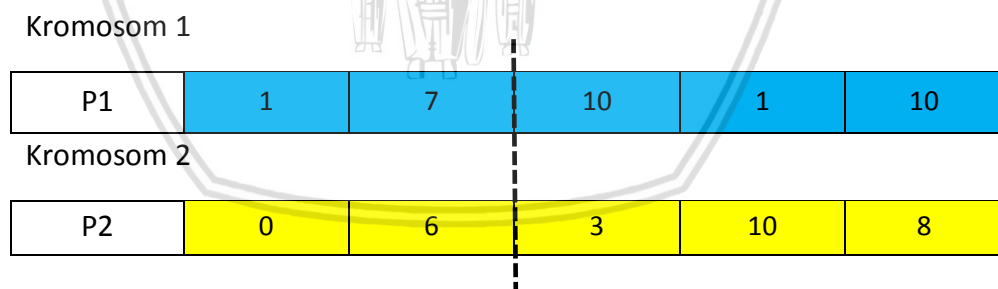


Gambar 3. 5 Flowchart Proses Crossover

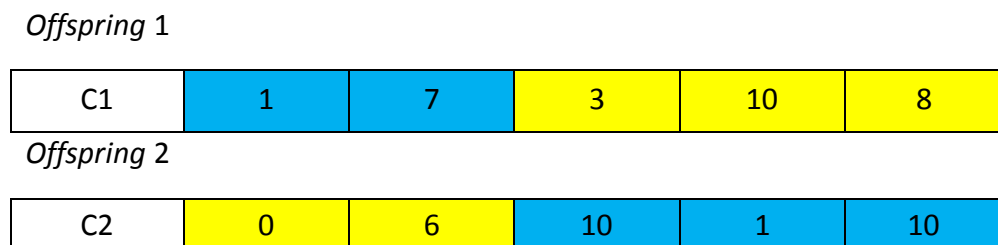


Gambar 3. 6 Flowchart Single-Point Crossover

Pada penelitian ini misalkan kita pilih kromosom *parent* yang terpilih yaitu P1 dan P2 untuk dilakukan *single-point crossover*. Titik potong terdapat pada titik ke 2.



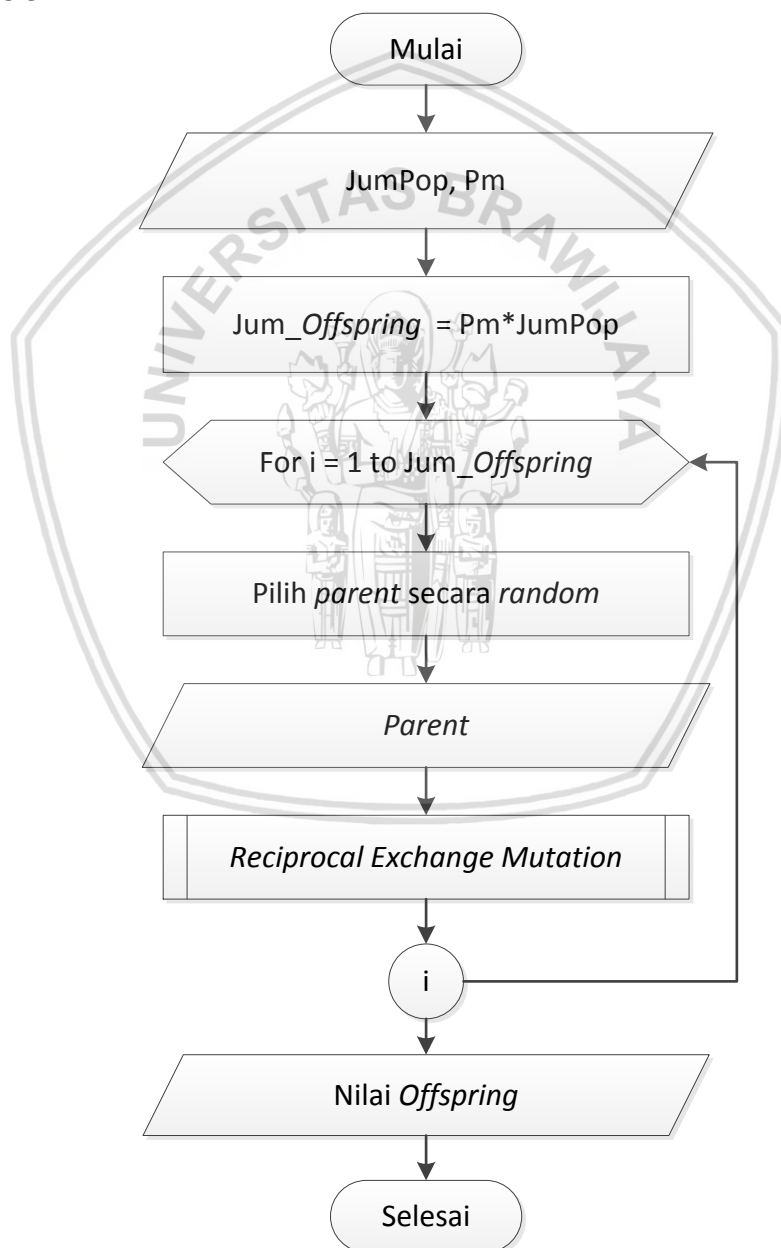
Maka, akan dihasilkan *Offspring* sebagai berikut



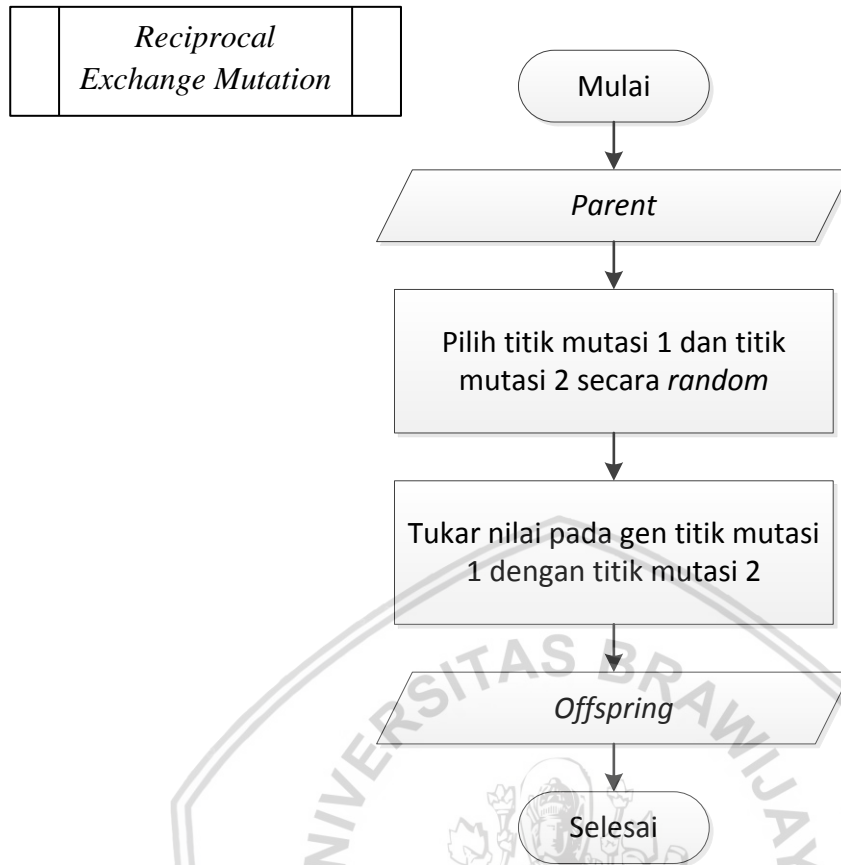
Gambar 3. 7 Single-Point Crossover

Setelah melakukan *crossover* maka dilakukan mutasi. Metode mutasi pada penelitian ini adalah *reciprocal exchange mutation*. Metode ini akan melakukan penukaran sebagian informasi dari 2 gen kromosom induk (*parent*) yang akan mengalami mutasi dan menghasilkan nilai *offspring*. Jumlah *offspring* yang akan dihasilkan diperoleh dari perkalian jumlah populasi dengan Probabilitas mutasi. Maka nilai *offspring* yang dihasilkan adalah $0,1 * 10 = 1 \text{ offspring}$.

Dalam pemilihan induk untuk melakukan mutasi dipilih secara acak. Setelah terpilih kromosom yang akan digunakan untuk proses mutasi. Kemudian pilih dua gen yang akan mengalami mutasi, setelah itu tukarkan informasi pada gen yang terpilih. Pada proses mutasi dapat dilihat pada Gambar 3.8 dan Gambar 3.9.



Gambar 3.8 Flowchart Proses Mutasi



Gambar 3.9 Flowchart Reciprocal Exchange Mutation

Pada penelitian ini, kromosom *random* yang terpilih misalkan kromosom 8 sebagai kromosom *parent* dengan titik tukar yaitu pada titik 2 dan titik ke 5.

Kromosom 8					
P8	0	9	3	2	0
Offspring					
C3	0	0	3	2	9

Gambar 3. 10 Reciprocal Exchange Mutation

- Setelah proses mutasi maka di dapatkan kumpulan populasi yang terdiri dari populasi awal dengan populasi *offspring* seperti ditunjuk pada Tabel 3.6.

Tabel 3.6 Tabel Populasi *Parent* dan *Offspring*

P(n)	K1	K2	K3	K4	K5	Total
P1	1	7	10	1	10	29
P2	0	6	3	10	8	27
P3	8	7	8	1	6	30
P4	3	9	2	1	1	16
P5	4	7	10	7	7	35
P6	4	10	10	1	1	26
P7	0	0	8	5	6	19
P8	0	9	3	2	0	14
P9	6	10	2	9	0	27
P10	4	5	1	5	6	21
C1	1	7	3	10	8	29
C2	0	6	10	1	10	27
C3	0	0	3	2	9	14

Untuk menghitung nilai *fitness*nya maka inisialisasi kromosom dalam bentuk integer dirubah dalam komposisi gram dengan rumus:

$$P_i = \frac{\text{Kromosom}[i]}{\text{Total}} \times \text{kemampuan konsumsi ayam per hari} \quad (3.1)$$

Misalkan pada P1 yaitu,

$$K1 = \frac{1}{29} \times 50 = 1.724137931$$

$$K2 = \frac{7}{29} \times 50 = 12.068966$$

$$K3 = \frac{10}{29} \times 50 = 17.24138$$

$$K4 = \frac{1}{29} \times 50 = 1.724137931$$

$$K5 = \frac{10}{29} \times 50 = 17.24138$$

Contoh lain pada perhitungan P2 yaitu,

$$K1 = \frac{0}{27} \times 50 = 0$$

$$K2 = \frac{6}{27} \times 50 = 11.111111$$

$$K3 = \frac{3}{27} \times 50 = 5.5555556$$

$$K4 = \frac{10}{27} \times 50 = 18.551852$$

$$K5 = \frac{8}{27} \times 50 = 14.81481$$

Contoh lain pada perhitungan P3 yaitu,

$$K1 = \frac{8}{30} \times 50 = 13.33333333$$

$$K2 = \frac{7}{30} \times 50 = 11.666667$$

$$K3 = \frac{8}{30} \times 50 = 13.33333$$

$$K4 = \frac{1}{30} \times 50 = 1.666667$$

$$K5 = \frac{6}{30} \times 50 = 10$$

Sesuai dengan Persamaan 3.1, maka hasil perhitungan komposisi dalam bentuk gram ditunjukkan pada Tabel 3.7 sebagai berikut.

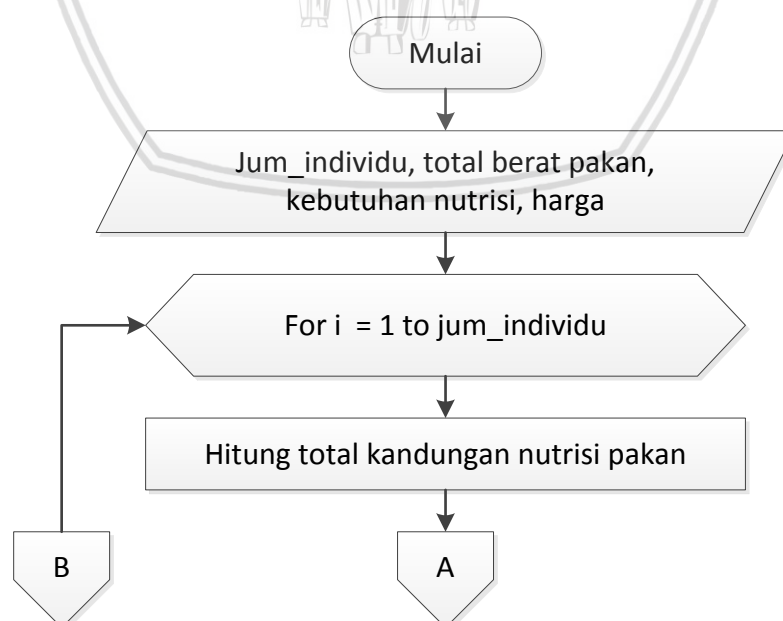
Tabel 3. 7 Komposisi Pakan Dalam Gram

P(n)	K1	K2	K3	K4	K5	Total
P1	1.724137931	12.068966	17.24138	1.724138	17.24138	50
P2	0	11.111111	5.555556	18.51852	14.81481	50
P3	13.33333333	11.666667	13.33333	1.666667	10	50
P4	9.375	28.125	6.25	3.125	3.125	50
P5	5.714285714	10	14.28571	10	10	50
P6	7.692307692	19.230769	19.23077	1.923077	1.923077	50
P7	0	0	21.05263	13.15789	15.78947	50
P8	0	32.142857	10.71429	7.142857	0	50
P9	11.11111111	18.518519	3.703704	16.66667	0	50
P10	9.523809524	11.904762	2.380952	11.90476	14.28571	50
C1	1.724137931	12.068966	5.172414	17.24138	13.7931	50
C2	0	11.111111	18.51852	1.851852	18.51852	50
C3	0	0	10.71429	7.142857	32.14286	50

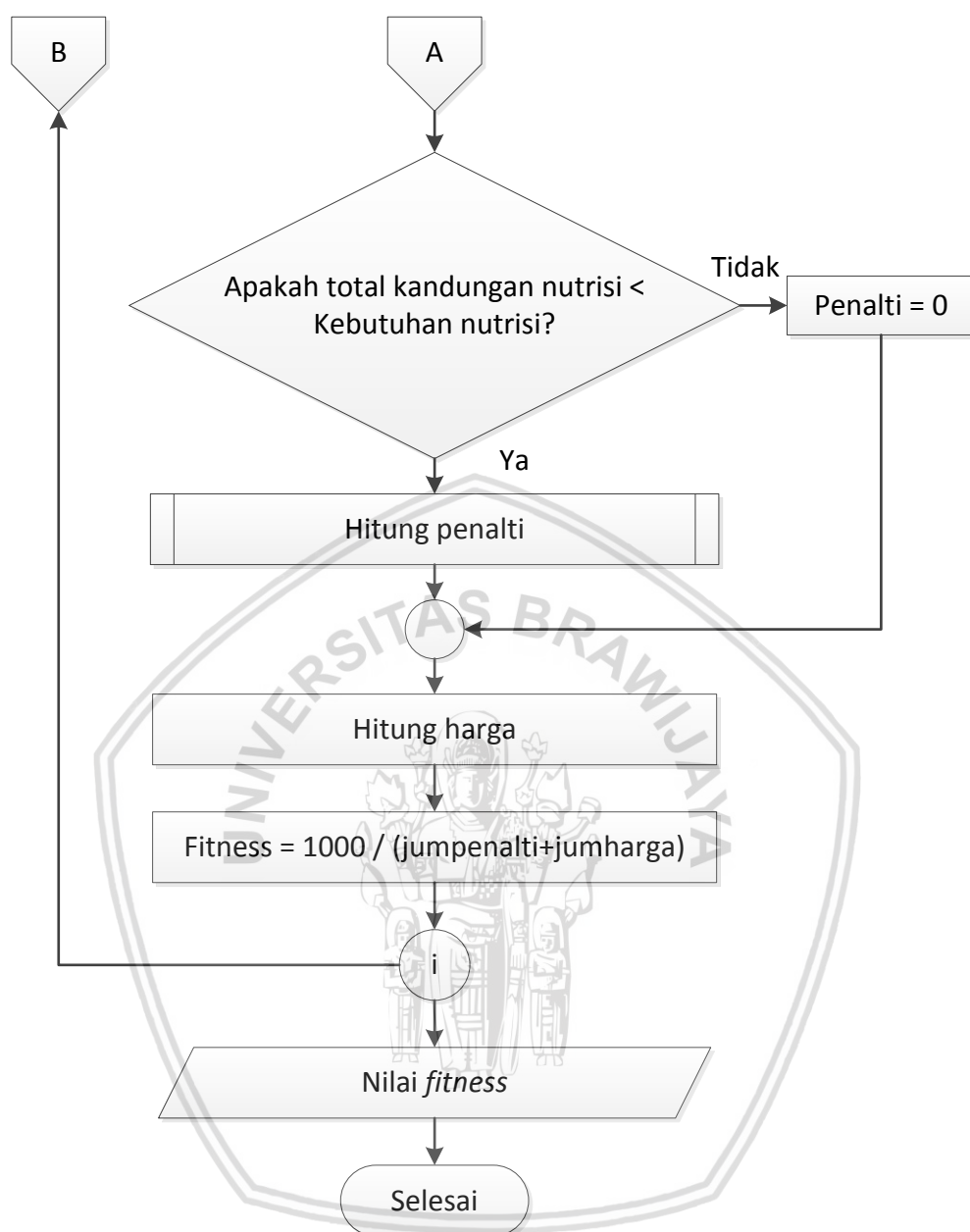
Setelah bobot diubah dalam bentuk gram, kemudian menghitung nutrisi yang terkandung sesuai dengan bobot sesuai dengan Persamaan (3.3) agar dapat menentukan apakah terkena penalti atau tidak. Perhitungan nutrisi pakan berdasarkan Persamaan 3.3 berikut.

$$\text{Nutrisi Pakan} = \sum (\text{Bobot Kromosom}[i] \times \text{NBp}) \quad (3.3)$$

Setelah kandungan nutrisi pada masing-masing individu dihitung, maka dihitung nilai *fitness*nya. Perhitungan nilai *fitness* berdasarkan Persamaan (2-2). *Flowchart* Perhitungan nilai *fitness* dapat dilihat pada Gambar 3.11 dan Gambar 3.12.

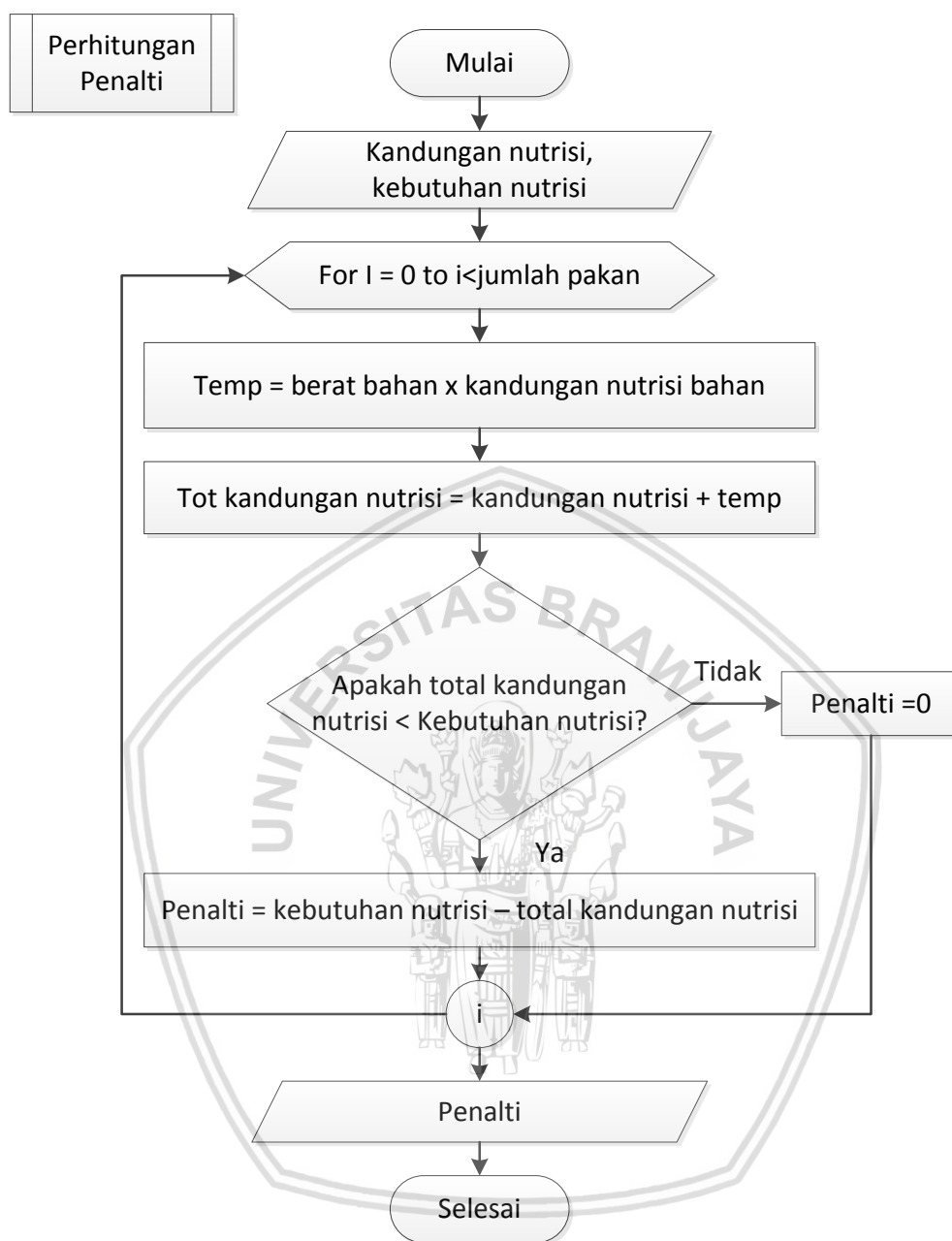


Gambar 3. 11 Flowchart Perhitungan Nilai Fitness



Gambar 3. 12 Flowchart Perhitungan Nilai *Fitness* (Lanjutan)

Pada penelitian optimasi komposisi pakan ayam petelur ini akan terjadi pinalti jika melanggar aturan. Aturan pinalti setiap kandungan nutrisi dapat dan persamaan perhitungan nilai pinalti dapat dilihat pada Persamaan (2-1). *Flowchart* perhitungan nilai pinalti yang ditunjukkan pada Gambar 3.13.



Gambar 3. 13 Flowchart Perhitungan Nilai Pinalti

Pada Tabel 3.8 menunjukkan kandungan nutrisi dalam bahan pakan terpilih dan harga pakan berdasarkan Tabel 3.1.

Tabel 3. 8 Tabel Kandungan Nutrisi Bahan Pakan Terpilih

Bahan pakan	Kandungan Nutrisi dalam bahan pakan					
	Em (Kkal/kg)	PK (%)	SK (%)	P (%)	CL (%)	Harga
Bekatul	2800	10	3	0.16	0.07	10
Beras pecah kulit	2660	8	9	0.04	0.09	21
Beras putih	3100	7.5	0.4	0.01	0.06	12
Dedak halus	2500	12	12	0.21	0.07	9.5
Dedak jagung	2950	10.6	5	0.15	0.07	7.8

Berikut perhitungan total kandungan nutrisi EM, PK, SK, P dan CL pada kromosom P1.

$$\text{Total Nutrisi} = \sum (\text{Bobot Kromosom (1)} \times N_{Bp(K1)}) + (\text{Bobot Kromosom (2)} \times N_{Bp(K2)}) + (\text{Bobot Kromosom (3)} \times N_{Bp(K3)}) + (\text{Bobot Kromosom (4)} \times N_{Bp(K4)}) + (\text{Bobot Kromosom (5)} \times N_{Bp(K5)})$$

$$\text{Total Nutrisi EM} = \sum (1.724137931 \times 2800) + (12.068966 \times 2660) + (17.24138 \times 3100) + (1.724138 \times 2500) + (17.24138 \times 2950)$$

$$\text{Total Nutrisi EM} = 145551.7241$$

$$\text{Total Nutrisi PK} = \sum (1.724137931 \times 10\%) + (12.068966 \times 8\%) + (17.24138 \times 7.5\%) + (1.724138 \times 12\%) + (17.24138 \times 10.6\%)$$

$$\text{Total Nutrisi PK} = 4.465517241$$

$$\text{Total Nutrisi SK} = \sum (1.724137931 \times 3\%) + (12.068966 \times 9\%) + (17.24138 \times 0.4\%) + (1.724138 \times 12\%) + (17.24138 \times 5\%)$$

$$\text{Total Nutrisi SK} = 2.275862069$$

$$\text{Total Nutrisi P} = \sum (1.724137931 \times 0.16\%) + (12.068966 \times 0.04\%) + (17.24138 \times 0.01\%) + (1.724138 \times 0.21\%) + (17.24138 \times 0.15\%)$$

$$\text{Total Nutrisi P} = 0.038793103$$

$$\text{Total Nutrisi CL} = \sum (1.724137931 \times 0.07\%) + (12.068966 \times 0.09\%) + (17.24138 \times 0.06\%) + (1.724138 \times 0.07\%) + (17.24138 \times 0.07\%)$$

$$\text{Total Nutrisi CL} = 0.035689655$$

$$\text{Total Harga} = \sum (1.724137931 \times 10) + (12.068966 \times 21) + (17.24138 \times 12) + (1.724138 \times 9.5) + (17.24138 \times 7.8)$$

$$\text{Total Harga} = 628.4482759$$

Dari hasil perhitungan total nutrisi dapat dihitung total penalti pada masing-masing kromosom berdasarkan Persamaan (2-1).

$$\text{Total Penalti} = \text{Penalti EM} + \text{Penalti PK} + \text{Penalti SK} + \text{Penalti P} + \text{Penalti CL}$$

$$\text{Total Penalti} = 0 + 5.034482759 + 0.224137931 + 0.361206897 + 0.039310345$$

$$\text{Total Penalti} = 5.659137931$$

Pada Tabel 3.9 merupakan hasil perhitungan penalti pada semua individu.

Tabel 3. 9 Hasil Perhitungan Penalty

P(n)	Penalti					Total Penalti	Harga
	EM	PK	SK	P	CI		
P1	0	5.03448	0.22414	0.36121	0.03931	5.659138	628.4483
P2	3222.22	4.40185	0	0.33389	0.03833	3226.996	591.4815
P3	0	4.97333	0.29667	0.35417	0.039	5.663167	632.1667
P4	2531.25	5.1375	0	0.36188	0.035	2536.784	813.4375
P5	0	4.79714	0	0.34943	0.03943	5.186	611.5714
P6	0	5.31538	0.13462	0.37115	0.03808	5.859231	744.8077
P7	0	4.66842	0.04737	0.34658	0.04211	5.104474	500.7895
P8	3428.57	5.26786	0	0.37107	0.03464	3434.245	871.4286
P9	6481.48	4.62963	0	0.33944	0.03667	6486.487	702.7778
P10	2380.95	4.47381	0	0.33333	0.03786	2385.797	598.3333
C1	3241.38	4.4431	0	0.335	0.0381	3246.196	604.1379
C2	0	5.03704	0.27778	0.36204	0.03963	5.716481	617.5926
C3	0	4.43214	0	0.33571	0.04107	4.808929	447.1429

Setelah nilai penalti didapatkan, maka dapat dihitung nilai Fitness berdasarkan Persamaan 2-2. Berikut perhitungan penalti pada kromosom P1.

$$Fitness = \frac{1000}{(Harga + Penalty)}$$

$$Fitness P1 = \frac{1000}{(628.4483 + 5.659138)} = 1.57702$$

$$Fitness P2 = \frac{1000}{(591.4815 + 3226.996)} = 0.261884$$

$$Fitness P3 = \frac{1000}{(632.1667 + 5.663167)} = 1.567816$$

$$Fitness P4 = \frac{1000}{(813.4375 + 2536.7848)} = 0.298488$$

$$Fitness P5 = \frac{1000}{(611.5714 + 5.186)} = 1.621383$$

$$Fitness P6 = \frac{1000}{(744.8077 + 5.859231)} = 1.332149$$

$$Fitness P7 = \frac{1000}{(500.7895 + 5.104474)} = 1.976699$$

$$Fitness P8 = \frac{1000}{(871.42863 + 3434.2458)} = 0.232252$$

$$Fitness P9 = \frac{1000}{(702.7778 + 6486.487)} = 0.139096$$

$$Fitness\ P10 = \frac{1000}{(598.3333 + 2385.797)} = 0.335106$$

$$Fitness\ C1 = \frac{1000}{(604.1379 + 3246.196)} = 0.259718$$

$$Fitness\ C2 = \frac{1000}{(617.5926 + 5.716481)} = 1.604341$$

$$Fitness\ C3 = \frac{1000}{(447.1429 + 4.808929)} = 2.212625$$

Tabel 3.10 merupakan hasil perhitungan nilai *fitness* pada semua individu.

Tabel 3. 10 Hasil Perhitungan *Fitness*

P(n)	K1	K2	K3	K4	K5	Fitness	Harga
P1	1	7	10	1	10	1.57702	628.4483
P2	0	6	3	10	8	0.261884	591.4815
P3	8	7	8	1	6	1.567816	632.1667
P4	3	9	2	1	1	0.298488	813.4375
P5	4	7	10	7	7	1.621383	611.5714
P6	4	10	10	1	1	1.332149	744.8077
P7	0	0	8	5	6	1.976699	500.7895
P8	0	9	3	2	0	0.232252	871.4286
P9	6	10	2	9	0	0.139096	702.7778
P10	4	5	1	5	6	0.335106	598.3333
C1	1	7	3	10	8	0.259718	604.1379
C2	0	6	10	1	10	1.604341	617.5926
C3	0	0	3	2	9	2.212625	447.1429

Kemudian, dilakukan seleksi *elitism* dengan mengambil sejumlah individu sesuai dengan *popsizenya* yaitu 10 individu terbaik. Sepuluh individu ini akan membentuk populasi baru untuk dihitung pada generasi selanjutnya. Untuk mendapatkan 10 individu terbaik dilihat nilai *fitnessnya*. Semakin besar nilai *fitness* maka semakin optimal individu tersebut. Tabel populasi baru ditunjukkan pada Tabel 3.11 Berikut.

Tabel 3.11 Tabel Populasi Baru

P(Asal)	P(n)	K1	K2	K3	K4	K5	Fitness	Harga
C3	P1	0	0	3	2	9	2.212625	447.1429
P7	P2	0	0	8	5	6	1.976699	500.7895
P5	P3	4	7	10	7	7	1.621383	611.5714
C2	P4	0	6	10	1	10	1.604341	617.5926
P1	P5	1	7	10	1	10	1.57702	628.4483
P3	P6	8	7	8	1	6	1.567816	632.1667
P6	P7	4	10	10	1	1	1.332149	744.8077
P10	P8	4	5	1	5	6	0.335106	598.3333
P4	P9	3	9	2	1	1	0.298488	813.4375
P2	P10	0	6	3	10	8	0.261884	591.4815

- Setelah melakukan evaluasi dan seleksi maka langkah selanjutnya adalah memilih kromosom terbaik berdasarkan nilai *fitness* yang terbesar. Dimana kromosom terbaik menunjukkan komposisi pakan yang terpenuhi tetapi dengan biaya yang minimal. Kromosom yang memiliki nilai *fitness* terbesar adalah pada kromosom asal C3 dengan nilai *fitness* 2.212625 dengan harga minimum 447.1429.

3.5 Implementasi

Implementasi aplikasi optimasi komposisi pakan ayam petelur menggunakan algoritme genetika dilakukan berdasarkan perancangan yang dibuat. Implementasi aplikasi pemetaan ini menggunakan bahasa pemrograman berorientasi objek yaitu Java.

3.6 Perancangan Pengujian

Perancangan pengujian bertujuan untuk melakukan pengujian terhadap aplikasi yang dibuat. Pengujian dilakukan untuk mengetahui kualitas solusi yang dihasilkan dari aplikasi. Kualitas solusi berhubungan dengan tingkat keoptimalan solusi yang dihasilkan. Tidak adanya metode tertentu untuk menentukan parameter algoritme genetika membuat perlu dilakukannya evaluasi aplikasi dengan uji coba masukan berupa generasi, populasi, nilai *cr*, dan nilai *mr* untuk mendapatkan parameter yang optimal. Selain itu perlu dilakukan pengujian kasus untuk membandingkan hasil optimasi komposisi pakan ayam petelur menggunakan algoritme genetika dengan pakan yang digunakan peternak.

BAB 4 PERANCANGAN

Bab perancangan ini akan membahas rancangan dalam pembuatan sistem optimasi komposisi pakan ayam petelur menggunakan algoritme genetika. Perancangan sistem pada penelitian ini meliputi perancangan antarmuka pengguna dan perancangan pengujian. Pada penelitian ini data yang dipakai tidak tersimpan di database sehingga tidak ada perancangan database.

4.1 Perancangan Antarmuka

Perancangan antarmuka ini dibuat untuk mewakili keadaan sebenarnya dari implementasi program yang akan dibangun. Implementasi program ini dibagi menjadi 5 halaman, yaitu: halaman *input* data, halaman inisialisasi populasi awal, halaman proses *crossover* dan mutasi, halaman evaluasi, dan halaman seleksi.

4.1.1 Halaman awal

Rancangan antarmuka pengguna dari aplikasi optimasi komposisi pakan ternak ayam petelur pada penelitian ini dapat dilihat pada Gambar 4.1. Tampilan antarmuka pengguna program terdiri dari 5 tab yaitu *home*, inisialisasi, *crossover* dan mutasi, evaluasi, dan seleksi.

1 OPTIMASI KOMPOSISI PAKAN AYAM PETELUR MENGGUNAKAN ALGORITME GENETIKA

2

HOME INISIALISASI CROSSOVER DAN MUTASI EVALUASI SELEKSI

FASE : 3 CR : 6

BANYAK AYAM: 4 MR : 7

POPULASI : 5 ITERASI: 8

PROSES 9

PILIH BAHAN PAKAN 10

Gambar 4. 1 Rancangan Antarmuka Halaman Awal

Bagian-bagian dari rancangan antarmuka pengguna pada Gambar 4.1 dijelaskan sebagai berikut:

1. Panel *header* yang berisi judul dari penelitian
2. Panel yang berfungsi untuk menampilkan halaman program yang terdiri dari 5 tab menu yaitu *home*, inisialisasi, *crossover* dan mutasi, evaluasi, dan seleksi. Menu berwarna biru menandakan menu tersebut aktif

3. *ComboBox* untuk memilih fase pertumbuhan yang terdiri atas fase *starter*, *grower*, *layer 1* dan *layer 2*
4. *Textfield* untuk memasukkan banyak ayam yang akan diberi pakan
5. *Textfield* untuk memberikan nilai pada parameter algoritme genetika yaitu banyaknya jumlah populasi yang akan dibangkitkan
6. *Textfield* untuk memberikan nilai pada parameter algoritme genetika yaitu probabilitas *crossover*
7. *Textfield* untuk memberikan nilai pada parameter algoritme genetika yaitu probabilitas mutasi
8. *Textfield* untuk memasukkan banyaknya iterasi yang diinginkan oleh pengguna
9. Tombol proses digunakan ketika menu masukan telah diisi dan akan diproses
10. Panel yang berisi bahan pakan apa saja yang bisa dipilih oleh pengguna. Panel ini nantinya akan berisi *checkbox* sebanyak 50 bahan pakan ayam

4.1.2 Halaman inisialisasi algoritme genetika

Halaman inisialisasi merupakan halaman yang menampilkan populasi awal yang di bangkitkan secara *random* dengan individu sebanyak populasi masukan. Rancangan antarmuka dari *tab* inisialisasi algoritme genetika ditunjukkan pada Gambar 4.2.

OPTIMASI KOMPOSISI PAKAN AYAM PETELUR MENGGUNAKAN ALGORITME GENETIKA

HOME INISIALISASI CROSSOVER DAN MUTASI EVALUASI SELEKSI

POPULASI :

ITERASI :

Individu	Kromosom

Gambar 4. 2 Rancangan Antarmuka *Tab* Proses Algoritme Genetika

Berikut adalah keterangan dari komponen-komponen pada halaman inisialisasi:

1. *Textfield* untuk menampilkan *popsiz*e dan iterasi yang dimasukkan oleh pengguna
2. Tabel untuk menampilkan populasi awal yang dibangkitkan secara *random*. Tabel terdiri dari kolom individu dan kolom kromosom sesuai dengan *popsiz*e dan bahan pakan yang ditentukan pengguna

4.1.3 Halaman *crossover* dan mutasi

Halaman *crossover* dan mutasi merupakan halaman yang menampilkan proses *crossover* dan mutasi untuk menghasilkan anak sebagai solusi baru. Rancangan antarmuka dari halaman *crossover* dan mutasi ditunjukkan pada Gambar 4.3

OPTIMASI KOMPOSISI PAKAN AYAM PETELUR MENGGUNAKAN ALGORITME GENETIKA

HOME INISIALISASI CROSSOVER DAN MUTASI EVALUASI SELEKSI

Offspring Crossover : (1)

Parent 1	Parent 2	Cut point	Child 1	Child 2

(2)

Offspring Mutasi : (3)

Parent	Mutation point 1	Mutation Point 2	Child

(4)

Gambar 4. 3 Rancangan Antarmuka *Tab Crossover* Halaman Reproduksi

Berikut keterangan dari komponen-komponen pada halaman *crossover* dan mutasi:

1. *Textfield* untuk menampilkan jumlah *offspring* dari *crossover* yang dilakukan
2. Tabel untuk menampilkan hasil *crossover* yang terdiri dari *parent 1*, *parent 2*, *cut point*, *child 1*, *child 2*
3. *Textfield* untuk menampilkan jumlah *offspring* dari mutasi yang dilakukan
4. Tabel untuk menampilkan hasil mutasi yang terdiri dari *parent*, *mutation point 1*, *mutation point 2*, *child*

4.1.4 Halaman evaluasi

Halaman evaluasi merupakan halaman yang menampilkan populasi sebanyak *popsiz* ditambah dengan individu baru dari hasil *crossover* dan mutasi yang kemudian dihitung nilai *fitness*nya. Rancangan antarmuka dari halaman evaluasi ditunjukkan pada Gambar 4.4.

OPTIMASI KOMPOSISI PAKAN AYAM PETELUR MENGGUNAKAN ALGORITME GENETIKA

HOME INISIALISASI CROSSOVER DAN MUTASI **EVALUASI** SELEKSI

POPULASI **1** :

ITERASI **2** :

Iterasi : **3** cr : **4** mr : **5**

Individu	Kromosom	Fitness
		6

Gambar 4. 4 Rancangan Antarmuka Halaman Evaluasi

Berikut keterangan dari komponen-komponen pada halaman evaluasi :

1. *Textfield* untuk menampilkan *popsiz* yang sudah dimasukkan pengguna
2. *Textfield* untuk menampilkan jumlah iterasi yang sudah dimasukkan pengguna
3. *ComboBox* untuk memilih iterasi ke berapa yang akan menampilkan hasil evaluasi
4. *Textfield* untuk menampilkan probabilitas *crossover*
5. *Textfield* untuk menampilkan probabilitas mutasi
6. Tabel untuk menampilkan hasil evaluasi yang terdiri dari individu, kromosom dan nilai *fitness*

4.1.5 Halaman seleksi

Halaman seleksi merupakan halaman yang akan menampilkan hasil dari proses seleksi menggunakan metode seleksi *elitism*. Dari hasil evaluasi di ambil individu dengan nilai *fitness* yang paling baik sebanyak *popsiz*. Rancangan antarmuka dari halaman seleksi ditunjukkan pada Gambar 4.5

OPTIMASI KOMPOSISI PAKAN AYAM PETELUR MENGGUNAKAN ALGORITME GENETIKA

HOME INISIALISASI CROSSOVER DAN MUTASI EVALUASI SELEKSI

ITERASI KE- : 1

Individu	Kromosom	Fitness	Harga

2

Individu Terbaik

Banyak ayam	Kromosom	Fitness	Total Harga

3

Gambar 4. 5 Rancangan Antarmuka Halaman Seleksi

Berikut keterangan dari komponen-komponen pada halaman seleksi :

1. *ComboBox* untuk memilih iterasi ke berapa yang akan menampilkan hasil dari seleksi
2. Tabel untuk menampilkan hasil seleksi yang terdiri dari individu terpilih, kromosom, nilai *fitness*, dan harga
3. Tabel untuk menampilkan individu terbaik dari semua iterasi yang terdiri dari banyak ayam, kromosom, nilai *fitness*, dan total harga

4.2 Perancangan Pengujian

Perancangan pengujian bertujuan untuk merancang pengujian apa saja yang akan diujikan terhadap aplikasi yang dibuat. Pengujian dilakukan untuk mengetahui kualitas solusi yang dihasilkan dari aplikasi. Kualitas solusi berhubungan dengan tingkat keoptimalan solusi yang dihasilkan. Tidak adanya metode tertentu untuk menentukan parameter Algoritme genetika membuat perlu dilakukannya evaluasi aplikasi dengan uji coba agar mendapatkan parameter yang optimal. Uji coba tersebut antara lain:

1. Uji coba untuk menentukan banyaknya generasi yang optimal
2. Uji coba untuk menentukan ukuran populasi yang optimal
3. Uji coba untuk mencari kombinasi probabilitas *crossover* (p_c) dan probabilitas mutasi (p_m) terbaik untuk menghasilkan rekomendasi yang optimal

4.2.1 Uji Coba Banyaknya Generasi

Uji coba banyaknya generasi yaitu uji coba yang digunakan untuk mengetahui ukuran generasi yang tepat agar dapat menghasilkan komposisi pakan ayam petelur yang optimal. Banyaknya generasi yang akan di uji yaitu antara 10-200. Rancangan uji coba banyak generasi dapat dilihat pada Tabel 4.1.

Tabel 4. 1 Rancangan Uji Coba Banyaknya Generasi

Banyak Generasi	Nilai <i>Fitness</i>										Rata- Rata <i>Fitness</i>
	Percobaan Pengujian ke-										
	1	2	3	4	5	6	7	8	9	10	
10											
20											
30											
40											
50											
60											
80											
100											
120											
140											
160											
180											
200											

4.2.2 Uji Coba Ukuran Populasi

Uji coba ukuran populasi yaitu uji coba yang digunakan untuk mengetahui ukuran populasi yang tepat agar dapat menghasilkan komposisi pakan ayam petelur yang optimal. Banyaknya populasi yang digunakan adalah kelipatan 50. Rancangan uji coba populasi dapat dilihat pada Tabel 4.2.

Tabel 4. 2 Rancangan Uji Coba Banyaknya Populasi

Banyak Populasi	Nilai <i>Fitness</i>										Rata-Rata <i>Fitness</i>
	Percobaan Pengujian ke-										
	1	2	3	4	5	6	7	8	9	10	
50											
100											
150											
200											
250											
300											
350											
400											
450											
500											
600											
700											
800											

4.2.3 Uji Coba Kombinasi P_c Dan P_m

Uji coba kombinasi probabilitas *crossover* (p_c) dan probabilitas mutasi (p_m) adalah uji coba yang dilakukan untuk mengetahui kombinasi p_c dan p_m terbaik untuk mendapatkan hasil komposisi pakan ayam petelur yang optimal. Uji coba ini menggunakan nilai p_c dan p_m yang berbeda antara *range* 0 - 1. Rancangan uji coba kombinasi probabilitas *crossover* dan probabilitas mutasi dapat dilihat pada Tabel 4.3

Tabel 4. 3 Rancangan Uji Coba Kombinasi P_c Dan P_m

Kombinasi		Nilai <i>Fitness</i>										Rata-Rata <i>Fitness</i>
		Percobaan Pengujian ke-										
P _c	P _m	1	2	3	4	5	6	7	8	9	10	
1	0											
0.8	0.1											
0.6	0.2											
0.5	0.3											
0.4	0.4											
0.3	0.5											
0.2	0.6											
0.1	0.8											
0	1											

BAB 5 IMPLEMENTASI

Bab implementasi akan dibahas tentang implementasi algoritme genetika pada pakan ayam petelur sesuai perancangan pada bab sebelumnya. Bab Implementasi terdiri dari beberapa sub-bab diantaranya yaitu spesifikasi aplikasi, batasan-batasan dalam implementasi, implementasi algoritme pada program, dan implementasi antarmuka.

5.1 Spesifikasi Aplikasi

Sub-bab spesifikasi aplikasi akan menjelaskan tentang spesifikasi sistem perancangan perangkat lunak yang menjadi rujukan untuk implementasi sistem. Spesifikasi sistem yang akan dibahas adalah spesifikasi perangkat keras dan spesifikasi perangkat lunak.

5.1.1 Spesifikasi Perangkat Keras

Spesifikasi perangkat keras untuk implementasi aplikasi optimasi komposisi pakan ayam petelur menggunakan algoritme genetika yaitu komputer dengan spesifikasi yang dipakai sebagai berikut.

- a. Prosesor Intel® Pentium® CPU 2020M @2.40GHz
- b. RAM 2 GB
- c. Harddisk dengan kapasitas 500 GB
- d. Monitor 14 inch

5.1.2 Spesifikasi Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan dalam implementasi aplikasi optimasi komposisi pakan ayam petelur menggunakan algoritme genetika ini antara lain sebagai berikut :

- a. Aplikasi operasi Windows 7 32-bit
- b. *Java Development Kit* (JDK) 7
- c. *Netbeans* IDE 8.0
- d. *Java library jexcelapi* 2.6.12 dan poi 2.9
- e. *Microsoft Excel* 2010

5.2 Batasan Implementasi

Batasan implementasi merupakan batasan sesuai perancangan yang dibuat agar penelitian memiliki ruang lingkup yang jelas. Batasan dalam implementasi optimasi komposisi pakan ayam petelur menggunakan algoritme genetika adalah sebagai berikut.

1. Aplikasi dibangun berdasarkan ruang lingkup *desktop application* dengan menggunakan bahasa pemrograman Java
2. Data-data yang digunakan pada aplikasi berasal dari data yang diimpor dari *file* Microsoft Excel dan masukan pengguna

3. Data yang diimpor dari *file* Microsoft Excel meliputi data kandungan gizi dan kebutuhan nutrisi pakan ayam petelur beserta dengan harga pakan
4. Masukan dari pengguna meliputi fase pertumbuhan ayam, banyak ayam, *popsi*, p_c , p_m , iterasi, dan bahan pakan ayam untuk proses optimasi komposisi pakan ayam petelur
5. Metode yang digunakan dalam menyelesaikan masalah optimasi komposisi pakan ayam petelur adalah metode algoritme genetika
6. Keluaran dari aplikasi berupa hasil komposisi bahan pakan ayam petelur yang dipilih oleh pengguna dimana kebutuhan nutrisi pakan ayam terpenuhi dengan harga yang minimal

5.3 Implementasi Algoritme

Pada sub-bab ini akan dijelaskan penerapan hasil perancangan kedalam sebuah program. Aplikasi optimasi komposisi pakan ayam petelur menggunakan algoritme genetika terdiri dari beberapa proses utama yang meliputi proses pembangkitan populasi awal, proses *crossover* dengan metode *single-point crossover*, proses mutasi dengan metode *exchange mutation*, proses perhitungan nilai *fitness* dan proses seleksi dengan metode seleksi *elitism*.

5.3.1 Implementasi Proses Pembangkitan Populasi Awal

Pembangkitan populasi awal adalah proses untuk membangkitkan populasi awal dengan *popsi* dan jumlah kromosom sebanyak bahan pakan yang dimasukkan pengguna secara random. Populasi awal yang dibangkitkan diberi bobot nilai 1-10 secara random. Implementasi proses pembangkitan populasi awal ditunjukkan pada *Source code* 5.1.

```

1 Random rdm = new Random ();
2 int nilaiRandom;
3 int k=Integer.parseInt(populasi.getText());
4 int k2=Pakan.size();
5 int[][] angka = new int[k][k2];
6 for (int i=0;i<k;i++){
7     for (int j=0;j<k2;j++){
8         nilaiRandom = rdm.nextInt (11);
9         angka [i][j]=nilaiRandom;
10    }
11 }
```

Source code 5. 1 Membangkitkan Populasi Awal

Pada *Source code* 5.1 menunjukkan bahwa :

1. Baris ke 3 untuk menentukan panjang populasi yang berasal dari masukan dari pengguna
2. Baris ke 4 untuk menentukan panjang kromosom dalam 1 individu sebanyak bahan pakan yang dimasukkan oleh pengguna
3. Baris ke 7-10 membangkitkan nilai *random* sebanyak populasi dengan panjang kromosom sebanyak bahan pakan yang dipilih

5.3.2 Implementasi Proses Crossover

Metode *crossover* yang digunakan yaitu metode *single-point crossover*. *Crossover* dilakukan dengan menyilangkan induk yang dipilih secara random sehingga menghasilkan *offspring/child*. Jumlah *offspring/child* yang dihasilkan sebanyak jumlah probabilitas *crossover* dikalikan dengan banyak populasi. Implementasi proses *crossover* ditunjukkan pada *Source code 5.2*.

```

1 .....
2 double ProbCros = Populasi*Cr;
3 int Pc = (int) ProbCros;
4 int jmlCrossover=0;
5 if(Pc % 2 == 0){
6     jmlCrossover = Pc;
7 }
8 else if (Pc % 2 == 1) {
9     jmlCrossover = Pc+1;
10 }
11 for(int j=0;j<(jmlCrossover/2);j++){
12     int parentsatu = random.nextInt(Populasi_Awal.size());
13     int parentdua = random.nextInt(Populasi_Awal.size());
14     if (parentsatu==parentdua){
15         parentdua = random.nextInt(Populasi_Awal.size());
16     }
17     for (int g = 0; g<Populasi_Awal.get(parentsatu).size() ; g++){
18         Penampung1.add(Populasi_Awal.get(parentsatu).get(g));
19     }
20     for (int h = 0; h<Populasi_Awal.get(parentdua).size() ; h++){
21         Penampung2.add(Populasi_Awal.get(parentdua).get(h));
22     }
23     ParentCrossover1.add(Penampung1);
24     ParentCrossover2.add(Penampung2);
25     int min = 1;
26     int max =(Pakan.size());
27     int cutPoint = random.nextInt(max-min)+min;
28     CutPoint.add(cutPoint);
29     for (int c=0;c<Penampung1.size();c++){
30         if (c< cutPoint){
31             Tempor.add(Penampung1.get(c));
32         }
33         else{
34             Tempor.add(Penampung2.get(c));
35         }
36         if (c< cutPoint){
37             Tempor1.add(Penampung2.get(c));
38         }
39         else{
40             Tempor1.add(Penampung1.get(c));
41         }
42     }

```

```

43 children1.add(Tempor);
44 children2.add(Tempor1);
45 }
46 //HILANGKAN 1 HASIL CROSSOVER UNTUK CROSSOVER GANJIL
47 if(Pc % 2 == 0){
48     children2 = children2;
49 }
50 else if (Pc % 2 == 1){
51     children2.remove((children2.size()-1));
52 }

```

Source code 5. 2 Proses Crossover

Pada *Source code 5.2* menunjukkan bahwa:

1. Baris ke 2 untuk menghitung banyak *offspring* yang akan dihasilkan
2. Baris ke 4-10 merupakan kondisi apabila nilai *offspring* ganjil dan genap
3. Baris 12-13 untuk menentukan indeks *parent crossover* secara *random*
4. Pada baris ke 14-16 merupakan kondisi jika indeks *parent* pertama sama dengan indeks *parent* kedua maka *parent* kedua akan melakukan *random* indeks lagi
5. Pada baris 17-26 untuk mengambil individu *parent* yang terpilih secara *random* dan menyimpannya ke *arraylist*
6. Baris ke 29 untuk mencari titik potong secara *random*
7. Baris ke 31-45 untuk proses *crossover*
8. Baris 50-55 memberikan kondisi jika nilai *crossover* genap maka hasil *offspring* tetap. Jika nilai *crossover* ganjil maka hasil *offspring* akan dihilangkan 1

5.3.3 Implementasi Proses Mutasi

Dalam penelitian ini digunakan metode *reciprocal exchange mutation*. Mutasi dilakukan dengan memilih induk secara *random* kemudian menukar posisi titik yang dipilih secara *random* juga sehingga menghasilkan *offspring/child*. Jumlah *offspring/child* yang dihasilkan sebanyak jumlah probabilitas mutasi dikalikan dengan banyak populasi. Implementasi proses mutasi ditunjukkan pada *Source code 5.3*.

```

1 .....
2 double ProbMut = Populasi*Mr;
3 int Pm = (int) ProbMut;
4 for(int j=0;j<Pm;j++){
5     int parentmutasi= random.nextInt(Populasi_Awal.size());
6     for (int m = 0; m<Populasi_Awal.get(parentmutasi).size() ; m++){
7         Penampungparent.add(Populasi_Awal.get(parentmutasi).get(m));
8         Penampunghasil.add(Populasi_Awal.get(parentmutasi).get(m));
9     }
10    Parentmutasi.add(Penampungparent);
11    int min = 0;
12    int max =(Pakan.size());
13    int Pointmutasi1 = random.nextInt(max-min)+min;

```

```

14 MutPoint1.add(Pointmutasi1);
15 int Pointmutasi2 = random.nextInt(max-min)+min;
16 if (Pointmutasi1==Pointmutasi2){
17     Pointmutasi2 = random.nextInt(max-min)+min;
18 }
19 MutPoint2.add(Pointmutasi2);
20 Collections.swap(Penampunghasil, Pointmutasi1, Pointmutasi2);
21 HasilMutasi.add(Penampunghasil);
22 }

```

Source code 5. 3 Proses Mutasi

Pada *Source code 5.3* menunjukkan bahwa:

1. Baris ke 2 untuk menentukan banyak *offspring* mutasi yang akan dihasilkan
2. Baris ke 4-11 untuk menentukan indeks *parent* mutasi secara *random* kemudian mengambil *parent* mutasi sesuai indeks yang terpilih dan menyimpannya kedalam *arraylist*
3. Baris ke 14-20 mengambil dua titik mutasi secara *random* dengan kondisi jika titik mutasi kedua sama dengan titik mutasi pertama maka akan dilakukan *random* lagi pada titik mutasi kedua
4. Baris ke 21-22 merupakan proses mutasi yaitu menukarkan titik mutase pertama dengan titik mutasi kedua

5.3.4 Implementasi Proses Perhitungan Penalti

Perhitungan kurang nutrisi atau penalti digunakan untuk menghitung kurangnya kandungan nutrisi dalam komposisi apabila total nutrisi dalam komposisi pakan kurang dari kebutuhan nutrisi yang dibutuhkan oleh ayam petelur. Penalti didapat dari penjumlahan dari selisih kebutuhan nutrisi dikurangi total kandungan nutrisi dalam komposisi pakan. Implementasi proses perhitungan kurang nutrisi ditunjukkan pada *Source code 5.4*.

```

1 .....
2 int panjangkromosom = (Populasi_Awal.size());
3 int banyakbahan =(Pakan.size());
4 for( n=0; n<panjangkromosom; n++){
5     totalgen = 0;
6     for( o=0;o<banyakbahan;o++){
7         totalgen = totalgen+Populasi_Awal.get(n).get(o);
8     }
9     TotalGen.add(totalgen);
10 }
11 for (int i=0;i<panjangkromosom;i++){
12     for (int j=0;j<banyakbahan;j++){
13         if (Populasi_Awal.get(i).get(j)==0){
14             beratpakan = 0;
15         }
16         else {
17             beratpakan = ((Populasi_Awal.get(i).get(j) *
18                 kebutuhanpakan) / TotalGen.get(i));
19         }
20         Bobotpakan [i][j] = beratpakan;

```



```

21     }
22 }
23 for( n=0; n<panjangkromosom; n++){
24     HitungEM =0;
25     HitungPK =0;
26     HitungSK =0;
27     HitungP =0;
28     HitungCl =0;
29     HitungHarga =0;
30     for( o=0;o<banyakbahan;o++){
31         HitungEM = HitungEM +
32             ((HasilBobot.get(n).get(o))*EM.get(o));
33         HitungPK = HitungPK +
34             ((HasilBobot.get(n).get(o))*PK.get(o))/100;
35         HitungSK = HitungSK +
36             ((HasilBobot.get(n).get(o))*SK.get(o))/100;
37         HitungP = HitungP +
38             ((HasilBobot.get(n).get(o))*P.get(o))/100;
39         HitungCl = HitungCl +
40             ((HasilBobot.get(n).get(o))*Cl.get(o))/100;
41         HitungHarga = HitungHarga +
42             ((HasilBobot.get(n).get(o))*Harga.get(o));
43     }
44     TotalEM.add(HitungEM);
45     TotalPK.add(HitungPK);
46     TotalSK.add(HitungSK);
47     TotalP.add(HitungP);
48     TotalCl.add(HitungCl);
49     TotalHarga.add(HitungHarga);
50 }
51 double penaltiEM = 0;
52 for( n=0; n<panjangkromosom; n++){
53     if (TotalEM.get(n) > batasEM){
54         penaltiEM = 0;
55     }
56     else if (TotalEM.get(n) < batasEM){
57         penaltiEM = batasEM-TotalEM.get(n);
58     }
59     PenaltiEM.add(penaltiEM);
60 }
61 double penaltiPK = 0;
62 for( n=0; n<panjangkromosom; n++){
63     if (TotalPK.get(n) > batasPK){
64         penaltiPK = 0;
65     }
66     else if (TotalPK.get(n) < batasPK){
67         penaltiPK = batasPK-TotalPK.get(n);
68     }
69     PenaltiPK.add(penaltiPK);
70 }
71 double penaltiSK = 0;
72 for( n=0; n<panjangkromosom; n++){
73     if (TotalSK.get(n) > batasSK){
74         penaltiSK = 0;
75     }
76     else if (TotalSK.get(n) < batasSK){
77         penaltiSK = batasSK-TotalSK.get(n);
78     }

```

```

79     PenaltiSK.add(penaltiSK);
80 }
81 double penaltiP = 0;
82 for( n=0; n<panjangkromosom; n++){
83     if (TotalP.get(n) > batasP){
84         penaltiP = 0;
85     }
86     else if (TotalP.get(n) < batasP){
87         penaltiP = batasP-TotalP.get(n);
88     }
89     PenaltiP.add(penaltiP);
90 }
91 double penaltiCl = 0;
92 for( n=0; n<panjangkromosom; n++){
93     if (TotalCl.get(n) > batasCl){
94         penaltiCl = 0;
95     }
96     else if (TotalCl.get(n) < batasCl){
97         penaltiCl = batasCl-TotalCl.get(n);
98     }
99     PenaltiCl.add(penaltiCl);
100 }
101 //TOTAL PENALTI
102 ArrayList<Double> TotalPenalti=new ArrayList<>();
103 TotalPenalti.clear();
104 double TotPenalti =0;
105 for( n=0; n<panjangkromosom; n++){
106     TotPenalti= PenaltiEM.get(n)+PenaltiPK.get(n)+
107                 PenaltiSK.get(n)+ PenaltiP.get(n)+
108                 PenaltiCl.get(n);
109     TotalPenalti.add(TotPenalti);
110 }

```

Source code 5.4 Proses Perhitungan Penalti

Pada *Source code 5.4* menunjukkan bahwa:

1. Baris 2-8 untuk menghitung total gen dalam satu individu
2. Baris 11-22 untuk mengubah nilai kromosom ke dalam bentuk gram
3. Baris 30-43 adalah perulangan untuk menghitung nilai kandungan nutrisi dan untuk menghitung harga pakan
4. Baris 52-101 untuk menentukan apakah nilai kandungan nutrisi masing-masing kurang dari kebutuhan nutrisi. Apabila nilai kandungan nutrisi kurang dari kebutuhan nutrisi maka akan dihitung penaltinya
5. Baris ke 103-111 untuk menghitung total penalti dari semua nutrisi

5.3.5 Implementasi Proses Perhitungan *Fitness*

Proses perhitungan *fitness* dilakukan dengan menggunakan rumus perhitungan *fitness* yang telah ditentukan sebelumnya kepada individu yang telah dibangkitkan. Perhitungan *fitness* di peroleh dari angka 1000 dibagi dengan penjumlahan total harga dengan total penalti keseluruhan nutrisi (EM,PK,SK,P,Cl) dari satu kromosom. Nilai 1000 diperoleh dari rata-rata harga bahan pakan. Sedangkan total harga didapatkan dari jumlah harga setiap bahan pakan yang ada pada satu kromosom tersebut. Nilai *fitness* akan digunakan untuk

menentukan baik tidaknya setiap individu. Implementasi proses perhitungan *fitness* ditunjukkan pada *Source code 5.5*.

```

1 double nilaifitness=0;
2 for( n=0; n<panjangkromosom; n++){
3     nilaifitness= (1000 / (TotalHarga.get(n)+
4                     TotalPenalti.get(n)));
5     FitnessKromosom.add(nilaifitness);
6 }

```

Source code 5. 5 Proses Perhitungan *Fitness*

Pada *Source code 5.5* adalah *code* untuk menghitung nilai *fitness* dengan rumus seperti Persamaan 2-2.

5.3.6 Implementasi proses seleksi

Metode seleksi yang digunakan yaitu metode *elitism*. Seleksi dilakukan dengan cara mengambil individu dengan nilai *fitness* terbaik sebanyak *popsiz*. Hasil populasi setelah seleksi akan disimpan dan digunakan untuk perhitungan pada generasi selanjutnya. Implementasi proses seleksi ditunjukkan pada *Source code 5.6*

```

1 double temp,temp2 ;
2 for (int m = 0; m <SortFitness.size(); m++) {
3     IndexFitness.add(m);
4 }
5 int tempI;
6 for (int i=0; i < SortFitness.size()-1;i++){
7     int index =i;
8     for (int j= i+1; j< SortFitness.size();j++)
9         if (SortFitness.get(j)> SortFitness.get(index))
10            index = j;
11     temp = SortFitness.get(index);
12     temp2 = SortFitness.get(i);
13     SortFitness.set(index, temp2);
14     SortFitness.set(i,temp);
15     //tukar index
16     tempI = IndexFitness.get(index);
17     IndexFitness.set(index, IndexFitness.get(i));
18     IndexFitness.set(i, tempI);
19 }
20 for(int i=0; i<IndexFitness.size();i++){
21     SortPopulasi.add(Populasi_Awal.get(IndexFitness.get(i)));
22     SortHarga.add(TotalHarga.get(IndexFitness.get(i)));
23 }
24 System.out.println("Populasi sorting= "+ SortPopulasi);
25 //SELEKSI (AMBIL SEJUMLAH POPULASI INPUTAN
26 int pop = Integer.parseInt(populasi.getText());
27 for (int i=0; i< pop ; i++){
28     Populasi_Baru.add(SortPopulasi.get(i));
29     FitnessPopBaru.add(SortFitness.get(i));
30     HargaPopBaru.add(SortHarga.get(i));
31 }

```

Source code 5. 6 Proses Seleksi

Pada *Source code* 5.6 menunjukkan bahwa:

1. Baris 6-18 untuk mengurutkan nilai *fitness* dari terbesar ke terkecil dan mengambil index *fitness* kemudian menyimpannya ke array baru
2. Baris 20-22 untuk mengurutkan populasi dan harga sesuai dengan index *fitness* yang sudah tersimpan di array baru
3. Baris 27-31 untuk proses seleksi yaitu dengan mengambil populasi sejumlah masukan

5.4 Implementasi Antarmuka

Implementasi antarmuka aplikasi optimasi komposisi pakan ayam petelur menggunakan algoritme genetika ini terdiri dari 5 halaman yaitu halaman awal, halaman inisialisasi, halaman *crossover* dan mutasi, halaman evaluasi, dan halaman seleksi.

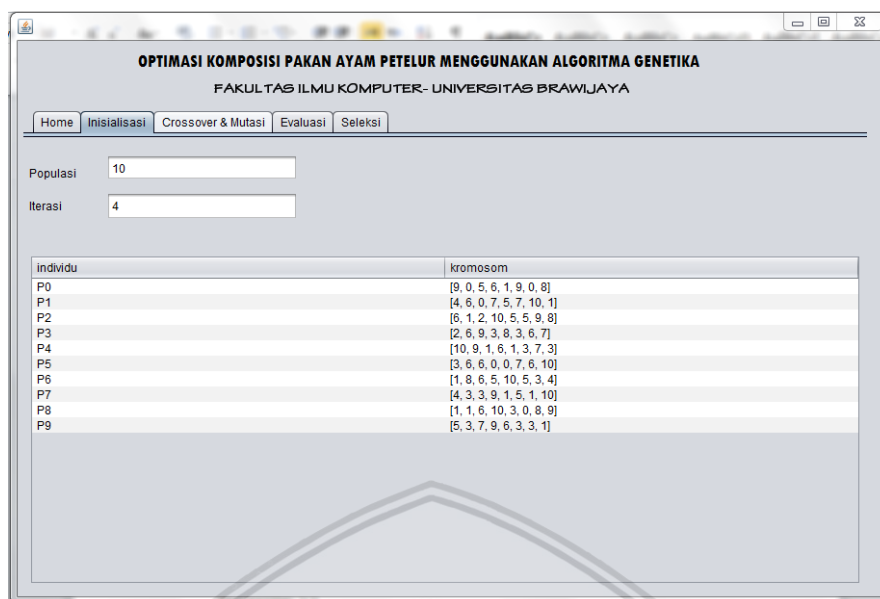
5.4.1 Halaman Awal

Pada halaman awal terdapat beberapa menu masukan yang harus diisi oleh pengguna. Menu masukan terdiri dari fase ayam, jumlah ayam, populasi, cr, mr, dan iterasi. Selain itu terdapat panel masukan jenis bahan pakan yang ingin dijadikan komposisi pakan oleh pengguna dalam bentuk *checkbox*. Tampilan halaman awal dapat dilihat pada Gambar 5.1.

Gambar 5. 1 Tampilan Halaman Awal

5.4.2 Halaman Inisialisasi

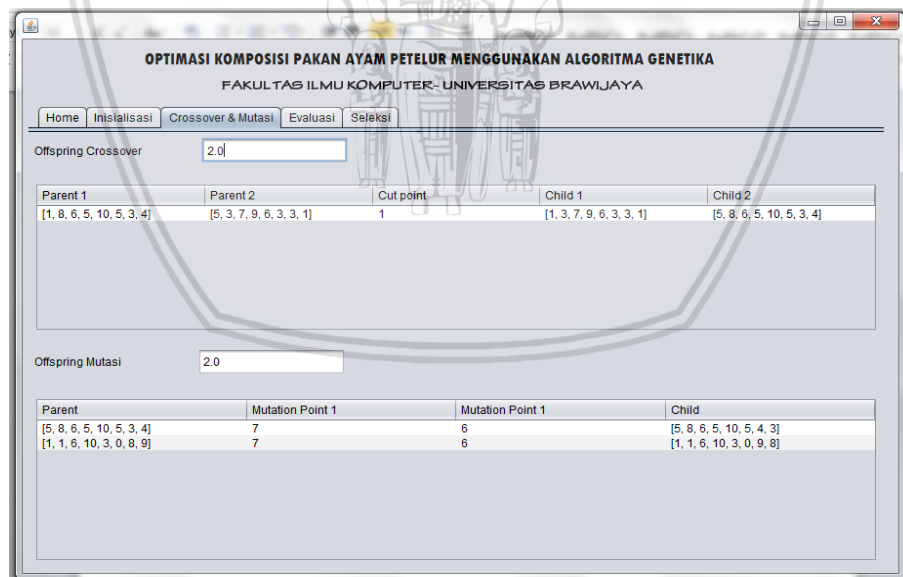
Setelah pengguna mengisi beberapa masukan pada halaman awal maka pada *tab* inisialisasi akan muncul populasi awal yang dibangkitkan secara *random* dengan jumlah individu sebanyak *popsiz*e dan kromosom sebanyak bahan pakan yang dipilih pengguna. Tampilan menu inisialisasi dapat dilihat pada Gambar 5.2.



Gambar 5. 2 Halaman Inisialisasi

5.4.3 Halaman Crossover dan Mutasi

Halaman *crossover* dan mutasi adalah halaman yang menampilkan proses *crossover* dan mutasi. Pada halaman ini terdapat dua tampilan tabel. Tabel pertama menampilkan proses *crossover* dan tabel kedua menampilkan proses mutasi. Halaman *crossover* dan mutasi dapat dilihat pada Gambar 5.3.



Gambar 5. 3 Halaman Crossover dan Mutasi

5.4.4 Halaman Evaluasi

Halaman evaluasi merupakan halaman yang menampilkan populasi sebanyak *popsiz*e ditambah dengan individu baru dari hasil proses *crossover* dan mutasi dan dilakukan proses perhitungan *fitness*. Pada halaman evaluasi juga terdapat pilihan iterasi dimana nanti akan ditampilkan proses evaluasi pada

iterasi yang dipilih pengguna. Tampilan untuk halaman evaluasi ditunjukkan pada Gambar 5.4.



OPTIMASI KOMPOSISI PAKAN AYAM PETELUR MENGGUNAKAN ALGORITMA GENETIKA
FAKULTAS ILMU KOMPUTER - UNIVERSITAS BRAWIJAYA

Home Inisialisasi Crossover & Mutasi **Evaluasi** Seleksi

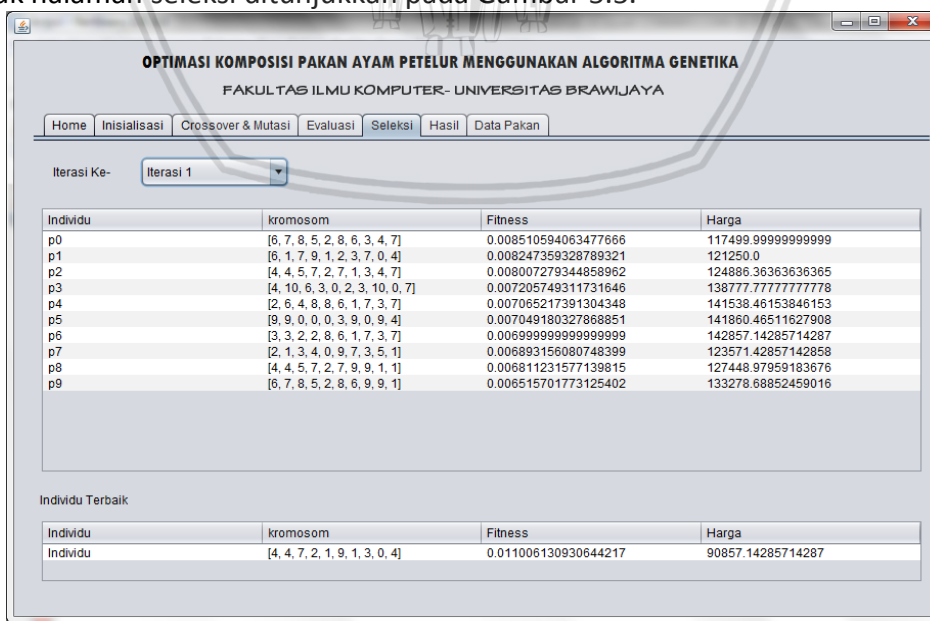
Populasi: 10
Iterasi: 4
Iterasi ke: Iterasi 1 Cr: 0.2 Mr: 0.2

Individu	kromosom	Fitness
p0	[9, 0, 5, 6, 1, 9, 0, 8]	0.012130175953660786
p1	[4, 6, 0, 7, 5, 7, 10, 1]	0.009702289844702965
p2	[6, 1, 2, 10, 5, 5, 9, 8]	0.01268000305367489
p3	[2, 6, 9, 3, 8, 3, 6, 7]	0.010640211682658614
p4	[10, 9, 1, 6, 1, 3, 7, 3]	0.010602397843067267
p5	[3, 6, 6, 0, 0, 7, 6, 10]	0.008645713179368525
p6	[1, 8, 6, 5, 10, 5, 3, 4]	0.010866121537064895
p7	[4, 3, 3, 9, 1, 5, 1, 10]	0.013054459284169533
p8	[1, 1, 6, 10, 3, 0, 8, 9]	0.013068537930841183
p9	[5, 3, 7, 9, 6, 3, 3, 1]	0.012713934410696071
p10	[1, 3, 7, 9, 6, 3, 3, 1]	0.011999241953343799
p11	[5, 8, 6, 5, 10, 5, 3, 4]	0.011427879750825964
p12	[5, 8, 6, 5, 10, 5, 4, 3]	0.011150844249144249
p13	[1, 1, 6, 10, 3, 0, 9, 8]	0.012634116255886165

Gambar 5. 4 Halaman Evaluasi

5.4.5 Halaman Seleksi

Halaman seleksi merupakan halaman yang akan menampilkan hasil dari proses perhitungan seleksi menggunakan metode seleksi *elitism* yaitu individu dengan nilai *fitness* terbaik akan terpilih sebanyak *popsiz*e. Pada halaman seleksi juga terdapat pilihan iterasi dimana nanti akan ditampilkan proses seleksi pada iterasi yang dipilih pengguna. Pada halaman seleksi juga ditampilkan individu terbaik dari setiap generasi dan individu terbaik dari semua generasi. Tampilan untuk halaman seleksi ditunjukkan pada Gambar 5.5.



OPTIMASI KOMPOSISI PAKAN AYAM PETELUR MENGGUNAKAN ALGORITMA GENETIKA
FAKULTAS ILMU KOMPUTER - UNIVERSITAS BRAWIJAYA

Home Inisialisasi Crossover & Mutasi Evaluasi **Seleksi** Hasil Data Pakan

Iterasi Ke-: Iterasi 1

Individu	kromosom	Fitness	Harga
p0	[6, 7, 8, 5, 2, 8, 6, 3, 4, 7]	0.008510594063477666	117499.99999999999
p1	[6, 1, 7, 9, 1, 2, 3, 7, 0, 4]	0.008247359328789321	121250.0
p2	[4, 4, 5, 7, 2, 7, 1, 3, 4, 7]	0.008007279344858962	124886.36363636365
p3	[4, 10, 6, 3, 0, 2, 3, 10, 0, 7]	0.007205749311731646	138777.77777777778
p4	[2, 6, 4, 8, 8, 6, 1, 7, 3, 7]	0.007065217391304348	141538.46153846153
p5	[9, 9, 0, 0, 0, 3, 9, 0, 9, 4]	0.007049180327868851	141860.46511627908
p6	[3, 3, 2, 2, 8, 6, 1, 7, 3, 7]	0.006999999999999999	142857.14285714287
p7	[2, 1, 3, 4, 0, 9, 7, 3, 5, 1]	0.006893156080748399	123571.42857142858
p8	[4, 4, 5, 7, 2, 7, 9, 9, 1, 1]	0.006811231577139815	127448.97959183676
p9	[6, 7, 8, 5, 2, 8, 6, 9, 9, 1]	0.006515701773125402	133278.68852459016

Individu Terbaik

Individu	kromosom	Fitness	Harga
Individu	[4, 4, 7, 2, 1, 9, 1, 3, 0, 4]	0.011006130930644217	90857.14285714287

Gambar 5. 5 Halaman Seleksi

BAB 6 PENGUJIAN

Bab ini membahas tentang pengujian dan analisa terhadap aplikasi optimasi komposisi pakan ayam petelur dengan metode algoritme genetika yang sudah diimplementasikan pada bab sebelumnya. Pengujian dan analisa dalam bab ini akan dibagi menjadi tiga sub bab yaitu sistematika pengujian dan analisis dan pembahasan.

6.1 Sistematika Pengujian

Berdasarkan rancangan pengujian pada bab 4, pengujian yang akan dilakukan yaitu pengujian generasi, populasi, kombinasi nilai pc dan pm, pengaruh nilai pc, pengaruh nilai pm dan studi kasus perbandingan hasil komputasi dengan hasil manual. Dari pengujian yang dilakukan dapat dilihat hasil komposisi yang paling optimal berdasarkan nilai *fitness*. Dalam melakukan pengujian ini, parameter tetap masukan yang dipakai adalah sebagai berikut:

- a. Fase ayam petelur yaitu fase *starter*
- b. Bahan pakan yang digunakan terdiri dari sepuluh bahan pakan yaitu bekatul, beras pecah kulit, beras putih, dedak halus, dedak jagung, gandum, gula, jagung kuning, menir dan pollard

6.2 Pengujian dan Pembahasan

Proses pengujian dilakukan sesuai perancangan uji coba yang telah dijelaskan pada Bab 4. Beberapa pengujian yang dilakukan pada penelitian ini yaitu pengujian banyaknya generasi, ukuran populasi, kombinasi nilai pc dan pm, pengaruh pc, pengaruh pm dan studi kasus. Pengujian ini menggunakan data bahan pakan yang ada pada Tabel 3.1 dan 3.2 dan data kebutuhan nutrisi ayam petelur pada Tabel 3.3.

6.2.1 Pengujian dan Analisa Banyaknya Generasi

Pengujian banyaknya generasi dilakukan untuk menentukan banyaknya generasi yang dapat menghasilkan solusi terbaik dalam optimasi komposisi pakan ayam petelur menggunakan algoritme genetika. Banyaknya generasi yang diuji adalah 10, 20, 30, 40, 50, 60, 80, 100, 120, 140, 160, 180, 200. Ukuran populasi yang digunakan adalah 160, probabilitas *crossover* 0.6 dan probabilitas mutasi 0.4. Parameter populasi, probabilitas *crossover* dan probabilitas mutasi didapatkan dari hasil nilai parameter terbaik dari penelitian sebelumnya oleh Ervina (2015). Untuk lebih detailnya mengenai parameter yang digunakan pada uji coba banyaknya generasi adalah sebagai berikut:

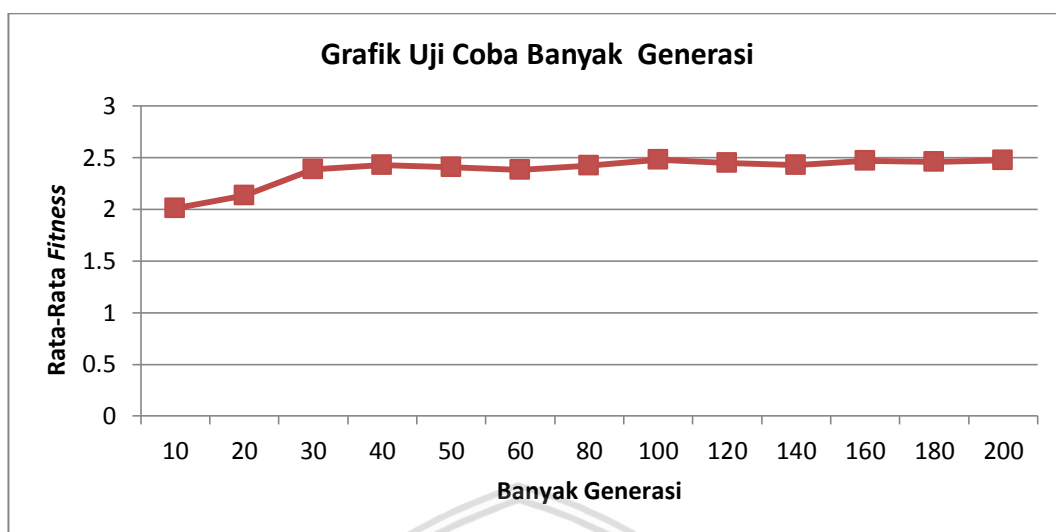
- | | |
|----------------------------------|----------|
| a. Ukuran populasi | : 160 |
| b. Banyaknya generasi | : 10-200 |
| c. Probabilitas <i>crossover</i> | : 0.6 |
| d. Probabilitas mutasi | : 0.4 |

Pengujian banyaknya generasi ini dilakukan sebanyak 10 kali kemudian dihitung nilai rata-rata *fitness* dan hasilnya dibandingkan dengan rata-rata *fitness* generasi lain. Dari pengujian banyak generasi tersebut diperoleh berapa ukuran generasi yang optimal untuk optimasi komposisi pakan ayam petelur. Hasil uji coba ukuran generasi dapat dilihat pada Tabel 6.1.

Tabel 6. 1 Hasil Uji Coba Banyaknya Generasi

Banyak Generasi	Nilai <i>Fitness</i>										Rata-Rata <i>Fitness</i>
	Percobaan Pengujian ke-										
	1	2	3	4	5	6	7	8	9	10	
10	1.9 929	2.0 360	2.0 027	2.0 422	2.0 014	1.9 906	1.9 616	2.0 170	2.0 043	2.0 490	2.00982
20	2.0 834	2.1 878	2.1 700	2.1 651	2.1 316	2.1 030	2.0 671	2.1 543	2.1 328	2.1 316	2.13272
30	2.4 479	2.2 333	2.4 479	2.2 676	2.4 479	2.3 076	2.5 901	2.2 213	2.5 901	2.3 366	2.38907
40	2.4 672	2.5 350	2.5 901	2.3 892	2.1 583	2.5 350	2.5 350	2.0 896	2.4 023	2.5 901	2.42924
50	2.2 593	2.1 089	2.1 499	2.4 402	2.2 610	2.5 901	2.5 901	2.5 907	2.5 350	2.5 350	2.40602
60	2.5 901	2.3 024	2.5 350	2.5 350	2.5 901	2.3 024	2.2 369	2.2 291	2.3 366	2.1 580	2.38161
80	2.3 024	2.5 350	2.3 024	2.5 901	2.2 179	2.5 350	2.3 024	2.5 350	2.5 901	2.3 024	2.42131
100	2.0 981	2.5 901	2.5 350	2.3 694	2.5 350	2.4 672	2.5 901	2.5 350	2.5 901	2.4 867	2.47973
120	2.5 350	2.0 981	2.5 901	2.4 867	2.5 901	2.5 350	2.5 350	2.2 593	2.3 024	2.5 350	2.44672
140	2.5 901	2.4 479	2.5 350	2.1 944	2.5 350	2.5 350	2.3 007	2.4 349	2.1 263	2.5 901	2.429
160	2.5 901	2.5 901	2.5 350	2.5 901	2.5 350	2.5 901	2.2 197	2.4 300	2.5 901	2.0 264	2.46973
180	2.3 007	2.3 024	2.5 901	2.5 350	2.5 901	2.5 350	2.5 901	2.3 024	2.3 024	2.5 350	2.45837
200	2.3 024	2.5 350	2.2 209	2.3 024	2.5 901	2.4 672	2.5 350	2.5 901	2.5 901	2.5 901	2.47239

Berdasarkan Tabel 6.1 dapat diketahui nilai rata-rata *fitness* setiap generasi. Untuk mempermudah melihat perbedaan rata-rata nilai *fitness* maka dibuatlah grafik rata-rata nilai *fitness* untuk uji coba banyaknya generasi. Grafik perbedaan hasil rata-rata nilai *fitness* setiap generasi dapat dilihat pada Gambar 6.1.



Gambar 6. 1 Grafik Hasil Uji Coba Banyaknya Generasi

Berdasarkan Gambar 6.1 dapat dilihat rata-rata nilai *fitness* mengalami peningkatan dari generasi 10 menuju generasi 30. Rata-rata nilai *fitness* yang dihasilkan pada generasi diatas 30 cenderung stabil karena perubahan rata-rata nilai *fitness* yang tidak begitu besar. Pada ukuran generasi 10, nilai rata-rata *fitness* 2,00982 yang merupakan nilai *fitness* terendah dari generasi lainnya. Sedangkan untuk *fitness* tertinggi terdapat pada generasi 100 yaitu 2,47973 yang merupakan ukuran generasi hasil yang optimal untuk menentukan komposisi pakan ayam petelur. Pada generasi 100 – 200 tidak mengalami kenaikan nilai rata-rata *fitness* yang signifikan melainkan mengalami konvergensi yang menunjukkan grafik hampir membentuk garis lurus. Hasil uji coba banyaknya generasi dengan parameter generasi antara 10-200 menunjukkan bahwa 100 adalah banyaknya generasi yang paling optimal. Dari uji coba banyaknya generasi, nilai *fitness* terbesar yang didapatkan yaitu 2.5901 yang muncul pertama kali pada generasi 30.

6.2.2 Pengujian dan Analisa Ukuran Populasi

Pengujian ukuran populasi digunakan untuk menentukan ukuran populasi yang tepat untuk menghasilkan solusi terbaik dalam kasus ini. Ukuran populasi yang akan diuji adalah 50, 100, 150, 200, 300, 350, 400, 450, 500, 600, 700 dan 800. Banyaknya generasi yang digunakan yaitu 100 yang didapat dari hasil coba banyaknya generasi pada percobaan sebelumnya. Probabilitas *crossover* dan probabilitas mutasi yang digunakan adalah 0.6 dan 0.4 yang didapatkan dari hasil nilai parameter terbaik dari penelitian sebelumnya oleh Ervina (2015). Untuk lebih detailnya nilai parameter yang digunakan pada uji coba populasi adalah sebagai berikut:

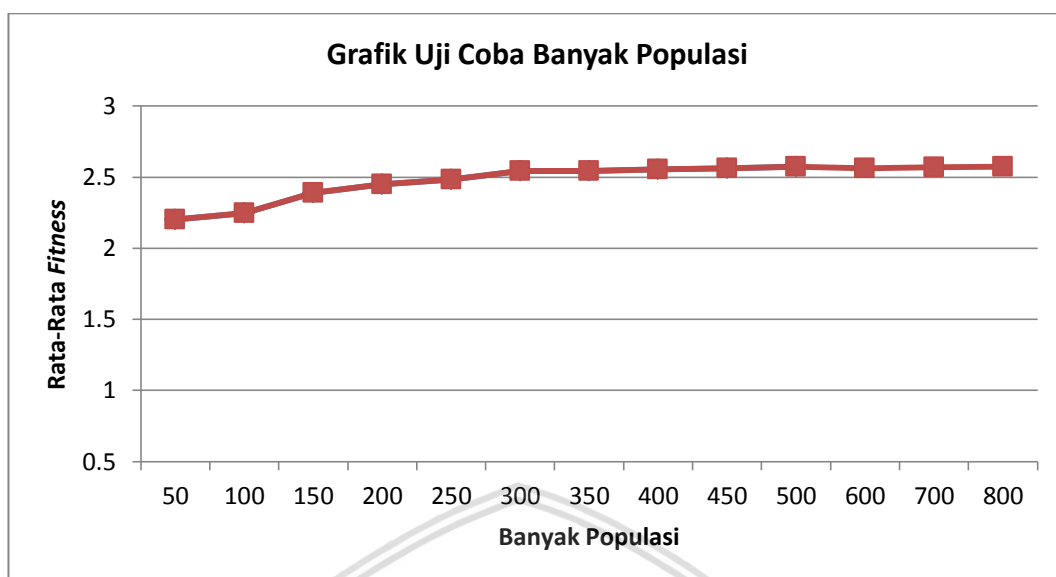
- a. Ukuran populasi : 50-500
- b. Banyaknya generasi : 100
- c. Probabilitas *crossover* : 0.6
- d. Probabilitas mutasi : 0.4

Pengujian banyaknya generasi ini dilakukan sebanyak 10 kali kemudian dihitung nilai rata-rata *fitness* dan hasilnya dibandingkan dengan rata-rata *fitness* populasi lain. Dari pengujian banyak populasi tersebut diperoleh berapa ukuran populasi yang optimal untuk optimasi komposisi pakan ayam petelur. Hasil uji coba ukuran populasi dapat dilihat pada Tabel 6.2.

Tabel 6. 2 Hasil Uji Coba Ukuran Populasi

Banyak Populasi	Nilai <i>Fitness</i>										Rata-Rata <i>Fitness</i>
	Percobaan Pengujian ke-										
	1	2	3	4	5	6	7	8	9	10	
50	2.0 830	2.0 494	2.5 900	2.2 900	2.0 596	2.0 379	2.4 301	1.9 910	2.4 479	2.0 334	2.201226
100	2.3 024	2.1 632	2.1 372	2.0 328	2.5 351	2.3 204	2.1 063	2.0 884	2.2 320	2.5 351	2.245289
150	2.5 351	2.5 901	2.2 138	2.5 901	2.5 901	2.3 024	2.5 351	2.2 314	2.0 971	2.2 127	2.389781
200	2.0 981	2.5 901	2.5 351	2.3 694	2.5 351	2.4 672	2.5 901	2.5 351	2.3 007	2.4 867	2.450754
250	2.3 024	2.2 134	2.5 901	2.5 901	2.5 351	2.5 901	2.5 901	2.5 351	2.5 901	2.3 024	2.483884
300	2.5 901	2.5 351	2.5 901	2.5 351	2.5 351	2.3 024	2.5 901	2.5 901	2.5 901	2.5 901	2.544821
350	2.5 351	2.5 901	2.5 901	2.5 901	2.5 351	2.5 901	2.5 901	2.3 024	2.5 351	2.5 901	2.544821
400	2.5 901	2.5 901	2.5 901	2.3 024	2.5 901	2.5 351	2.5 901	2.5 901	2.5 901	2.5 901	2.555827
450	2.5 901	2.5 351	2.5 901	2.5 901	2.5 351	2.5 901	2.5 351	2.5 901	2.5 901	2.4 672	2.561301
500	2.5 901	2.5 901	2.5 351	2.5 901	2.5 901	2.5 351	2.5 901	2.5 901	2.5 351	2.5 901	2.573591
600	2.5 901	2.5 351	2.5 351	2.5 901	2.5 901	2.5 351	2.5 351	2.5 901	2.5 901	2.5 351	2.56259
700	2.5 901	2.5 351	2.5 901	2.5 901	2.5 351	2.5 351	2.5 901	2.5 901	2.5 351	2.5 901	2.56809
800	2.5 901	2.5 901	2.5 901	2.5 351	2.5 351	2.5 351	2.5 901	2.5 901	2.5 901	2.5 901	2.573591

Berdasarkan Tabel 6.2 dapat diketahui nilai rata-rata *fitness* setiap populasi. Untuk mempermudah melihat perbedaan rata-rata nilai *fitness* maka dibuatlah grafik rata-rata nilai *fitness* untuk uji coba banyaknya populasi. Grafik perbedaan hasil rata-rata nilai *fitness* setiap populasi dapat dilihat pada Gambar 6.2.



Gambar 6. 2 Grafik Hasil Uji Coba Ukuran Populasi

Berdasarkan Gambar 6.2 dapat dilihat rata-rata nilai *fitness* mengalami peningkatan dari populasi 50 ke populasi 300. Rata-rata nilai *fitness* yang dihasilkan pada generasi diatas 300 cenderung stabil karena perubahan rata-rata nilai *fitness* yang tidak begitu besar. Pada ukuran populasi 50, nilai rata-rata *fitness* 2,201226 yang merupakan nilai *fitness* terendah dari populasi lainnya. Sedangkan untuk *fitness* tertinggi terdapat pada populasi 500 dan 800 yaitu 2,573591. Pada populasi 300 – 800 tidak mengalami kenaikan nilai rata-rata *fitness* yang signifikan melainkan mengalami konvergensi yang menunjukkan grafik hampir membentuk garis lurus. Hasil uji coba ukuran populasi dengan parameter populasi antara 50-800 menunjukkan bahwa 500 dan 800 adalah ukuran populasi yang paling optimal. Dari uji coba ukuran populasi, nilai *fitness* terbesar yang didapatkan yaitu 2.5901 yang muncul pertama kali pada populasi 150.

6.2.3 Pengujian dan Analisa Kombinasi Pc dan Pm

Pengujian kombinasi Pc dan Pm digunakan untuk menentukan kombinasi Pc dan Pm yang tepat untuk menghasilkan solusi terbaik dalam kasus ini. Kombinasi Pc dan Pm yang akan diuji yaitu antara 0 - 1. Untuk lebih detailnya nilai parameter yang digunakan pada uji coba kombinasi Pc dan Pm adalah sebagai berikut:

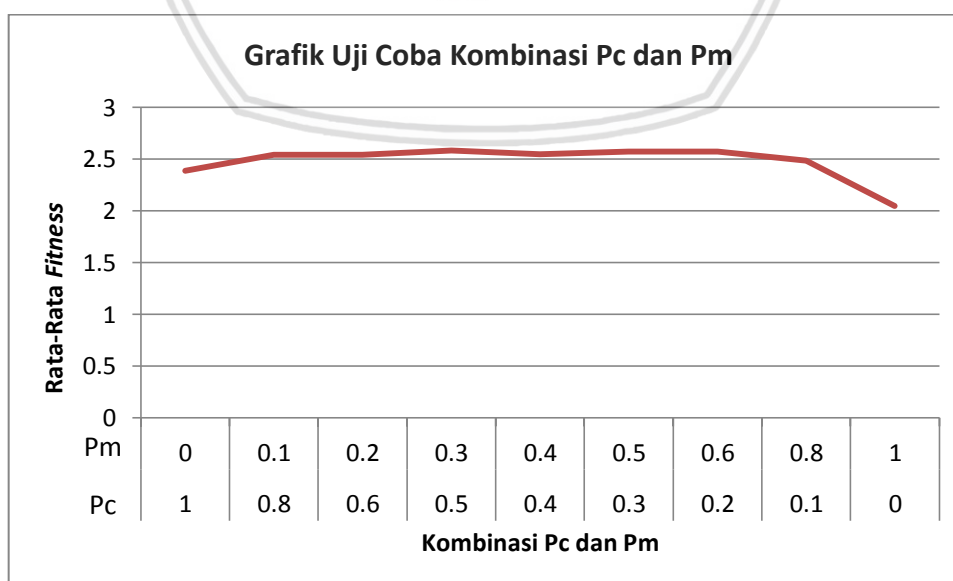
- a. Ukuran populasi : 500
- b. Banyaknya generasi : 100

Pengujian kombinasi Pc dan Pm ini dilakukan sebanyak 10 kali kemudian dihitung nilai rata-rata *fitness* dan hasilnya dibandingkan dengan rata-rata *fitness* generasi lain. Dari pengujian kombinasi pc dan pm tersebut diperoleh berapa kombinasi Pc dan Pm yang optimal untuk optimasi komposisi pakan ayam petelur. Hasil uji coba kombinasi Pc dan Pm dapat dilihat pada Tabel 6.3.

Tabel 6. 3 Hasil Uji Coba Kombinasi Pc dan Pm

Kombinasi		Nilai <i>Fitness</i>										Rata-Rata <i>Fitness</i>
		Percobaan Pengujian ke-										
Pc	Pm	1	2	3	4	5	6	7	8	9	10	
1	0	2.5 25	2.2 59	2.4 14	2.4 48	2.5 25	2.4 23	2.2 98	2.4 48	2.3 18	2.2 13	2.38706
0.8	0.1	2.5 90	2.5 90	2.5 90	2.5 90	2.5 90	2.5 35	2.5 90	2.3 02	2.5 35	2.5 35	2.5448
0.6	0.2	2.5 90	2.5 90	2.3 02	2.5 90	2.5 90	2.5 90	2.5 90	2.5 35	2.5 35	2.5 35	2.5448
0.5	0.3	2.5 90	2.5 90	2.5 90	2.5 90	2.5 90	2.5 90	2.5 90	2.5 35	2.5 90	2.5 90	2.58459
0.4	0.4	2.5 90	2.5 90	2.5 90	2.5 35	2.5 90	2.3 02	2.5 35	2.5 90	2.5 90	2.5 90	2.55031
0.3	0.5	2.5 90	2.5 35	2.5 35	2.5 90	2.5 90	2.5 90	2.5 35	2.5 90	2.5 90	2.5 90	2.57357
0.2	0.6	2.5 35	2.5 90	2.5 35	2.5 90	2.5 90	2.5 90	2.5 35	2.5 90	2.5 90	2.5 90	2.57357
0.1	0.8	2.3 02	2.3 02	2.3 01	2.5 90	2.5 35	2.5 90	2.5 90	2.5 35	2.5 35	2.5 90	2.48709
0	1	2.0 69	2.0 85	2.1 08	2.0 16	2.0 16	2.0 06	2.0 25	2.0 38	2.0 58	2.0 36	2.04557

Berdasarkan Tabel 6.3 dapat diketahui nilai rata-rata *fitness* setiap percobaan kombinasi Pc dan Pm. Untuk mempermudah melihat perbedaan rata-rata nilai *fitness* maka dibuatlah grafik rata-rata nilai *fitness* untuk uji coba kombinasi Pc dan Pm. Grafik perbedaan hasil rata-rata nilai *fitness* kombinasi Pc dan Pm dapat dilihat pada Gambar 6.3.



Gambar 6. 3 Grafik Hasil Uji Coba Kombinasi Pc dan Pm

Berdasarkan Gambar 6.3 dapat dilihat rata-rata nilai *fitness* mengalami penurunan pada kombinasi Pc 0 dan Pm 1. Penurunan rata-rata nilai *fitness* juga terjadi pada kombinasi Pc 1 dan Pm 0. Rata-rata nilai *fitness* yang dihasilkan pada kombinasi Pc lebih dari nol dan Pm lebih dari nol cenderung stabil karena tidak mengalami perubahan yang signifikan. Hasil uji coba ukuran populasi menunjukkan bahwa kombinasi Pc 0.5 dan Pm 0.3 adalah kombinasi Pc dan Pm yang optimal dengan rata-rata nilai *fitness* tertinggi yaitu 2.58459. Dari uji coba ukuran kombinasi pc dan pm, nilai *fitness* terbesar yang didapatkan yaitu 2.5901 yang muncul pada hasil uji coba kombinasi Pc lebih dari nol dan Pm lebih dari nol. Hal ini menunjukkan bahwa solusi terbaik lebih cepat muncul apabila probabilitas crossover dan mutasi bernilai lebih dari 0.

6.2.4 Pengujian dan Analisa Pengaruh Nilai Pm

Pengujian nilai Pm digunakan untuk mengetahui pengaruh nilai Pm yang bisa menghasilkan solusi terbaik dalam kasus ini. Nilai Pm yang akan diuji yaitu antara 0-1. Parameter populasi, dan generasi didapatkan pada pengujian yang sudah dilakukan sebelumnya. Nilai Pc yang digunakan pada pengujian ini yaitu 0.4. Untuk lebih detailnya nilai parameter yang digunakan pada uji coba pengaruh nilai Pc adalah sebagai berikut:

- Ukuran populasi : 500
- Banyaknya generasi : 100
- Probabilitas *crossover* : 0.4
- Probabilitas mutasi : 0-1

Pengujian dilakukan 10 kali dengan tujuan untuk mendapatkan hasil yang mewakili kemampuan algoritme secara keseluruhan. Nilai *fitness* pada setiap percobaan dihitung rata-ratanya untuk mengetahui pengaruh nilai Pm terhadap nilai *fitness*. Hasil uji coba pengaruh nilai Pm dapat dilihat pada Tabel 6.4 dan Tabel 6.5.

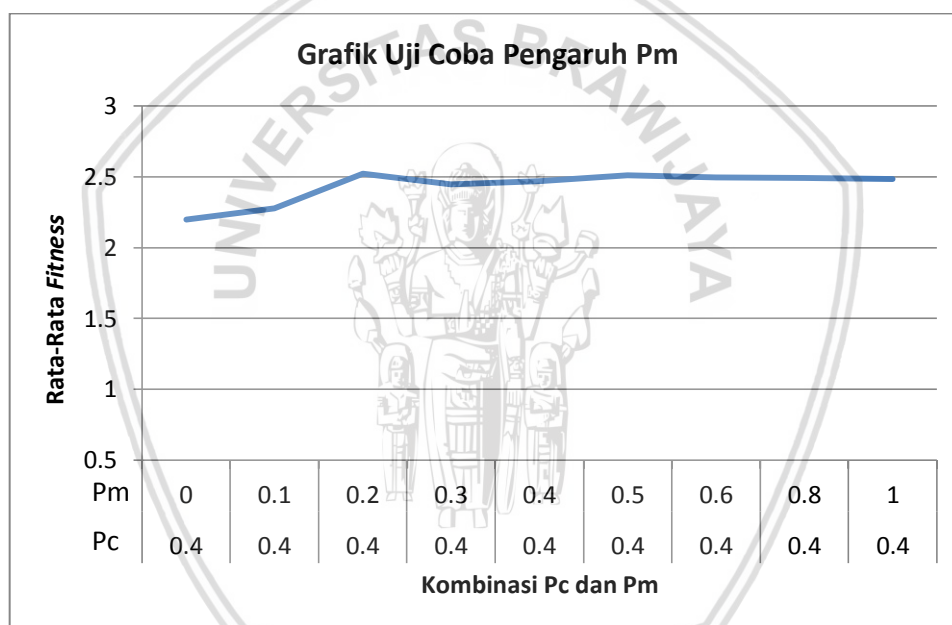
Tabel 6. 4 Hasil Uji Coba Pengaruh nilai Pm

Kombinasi		Nilai <i>Fitness</i>										Rata- Rata <i>Fitness</i>
		Percobaan Pengujian ke-										
Pc	Pm	1	2	3	4	5	6	7	8	9	10	
0.4	0	2.3 732	2.4 16	2.1 511	2.1 142	2.2 142	2.1 844	2.0 233	2.1 672	2.2 277	2.1 095	2.19808
0.4	0.1	2.2 024	2.1 502	2.4 16	2.0 964	2.5 902	2.2 893	2.3 083	2.2 793	2.1 743	2.2 793	2.27856
0.4	0.2	2.5 902	2.5 902	2.3 987	2.5 902	2.5 351	2.2 878	2.5 902	2.5 902	2.5 351	2.5 351	2.52426
0.4	0.3	2.3 024	2.2 857	2.5 902	2.5 902	2.3 024	2.5 902	2.5 902	2.5 902	2.5 902	2.0 458	2.44773
0.4	0.4	2.4 868	2.3 008	2.5 902	2.5 902	2.1 62	2.5 351	2.5 902	2.5 351	2.5 902	2.3 024	2.46828
0.4	0.5	2.5 902	2.4 672	2.5 902	2.3 008	2.5 902	2.2 847	2.5 351	2.5 902	2.5 902	2.5 902	2.51288
0.4	0.6	2.5 902	2.3 008	2.5 351	2.5 351	2.5 902	2.5 351	2.5 902	2.5 902	2.1 755	2.5 351	2.4977 2

Tabel 6. 5 Hasil Uji Coba Pengaruh nilai Pm (Lanjutan)

Kombinasi		Nilai <i>Fitness</i>										Rata- Rata <i>Fitness</i>
		Percobaan Pengujian ke-										
Pc	Pm	1	2	3	4	5	6	7	8	9	10	
0.4	0.8	2.5 902	2.5 351	2.5 902	2.5 902	2.5 902	2.5 902	2.3 024	2.3 024	2.3 024	2.5 351	2.49282
0.4	1	2.3 024	2.5 351	2.5 902	2.5 351	2.5 902	2.5 351	2.3 024	2.3 694	2.5 351	2.5 351	2.48299

Berdasarkan Tabel 6.4 dapat diketahui nilai rata-rata *fitness* setiap percobaan kombinasi Pc 0.4 dan Pm 0-1. Untuk mempermudah melihat perbedaan rata-rata nilai *fitness* maka dibuatlah grafik rata-rata nilai *fitness* untuk uji coba pengaruh nilai Pm. Grafik pengaruh nilai Pm terhadap nilai *fitness* dapat dilihat pada Gambar 6.4.



Gambar 6. 4 Grafik Hasil Uji Coba Pengaruh Nilai Pm

Berdasarkan Gambar 6.4 dapat dilihat rata-rata nilai *fitness* pada kombinasi pc 0.4 dan pm 0 merupakan rata-rata *fitness* terendah dari pada kombinasi pc dan pm lainnya. Rata-rata nilai *fitness* pada kombinasi pc 0.4 dan pm 0 adalah 2.19808. Kemudian pada kombinasi pc 0.4 dan pm 0.1 rata-rata nilai *fitness* mengalami kenaikan dengan nilai 2.27856. Pada kombinasi pc 0.4 dan pm 0.2 rata-rata nilai *fitness* kembali mengalami kenaikan menjadi 2.52426. Pada kombinasi pc 0.4 dan pm 0.3 rata-rata nilai *fitness* turun menjadi 2.44773 tetapi kemudian pada kombinasi pc 0.4 dan pm 0.4 mengalami kenaikan menjadi 2.44773. Setelah kombinasi pc 0.4 dan pm 0.5 rata-rata *fitness* cenderung stabil karena tidak mengalami perubahan yang signifikan. Berdasarkan tabel 6.4 nilai *fitness* terbesar yang didapatkan yaitu 2.5902 yang muncul pada hasil uji coba

kombinasi P_c 0.4 dan P_m lebih dari nol. Hal ini menunjukkan bahwa solusi terbaik lebih cepat muncul apabila probabilitas mutasi bernilai lebih dari 0.

6.2.5 Pengujian dan Analisa Pengaruh Nilai P_c

Pengujian nilai P_c digunakan untuk menentukan pengaruh nilai P_c untuk menghasilkan solusi terbaik dalam kasus ini. Nilai P_c yang akan diuji yaitu antara 0-1. Parameter populasi dan generasi didapatkan pada pengujian yang sudah dilakukan sebelumnya. Untuk lebih detailnya nilai parameter yang digunakan pada uji coba pengaruh nilai P_c adalah sebagai berikut:

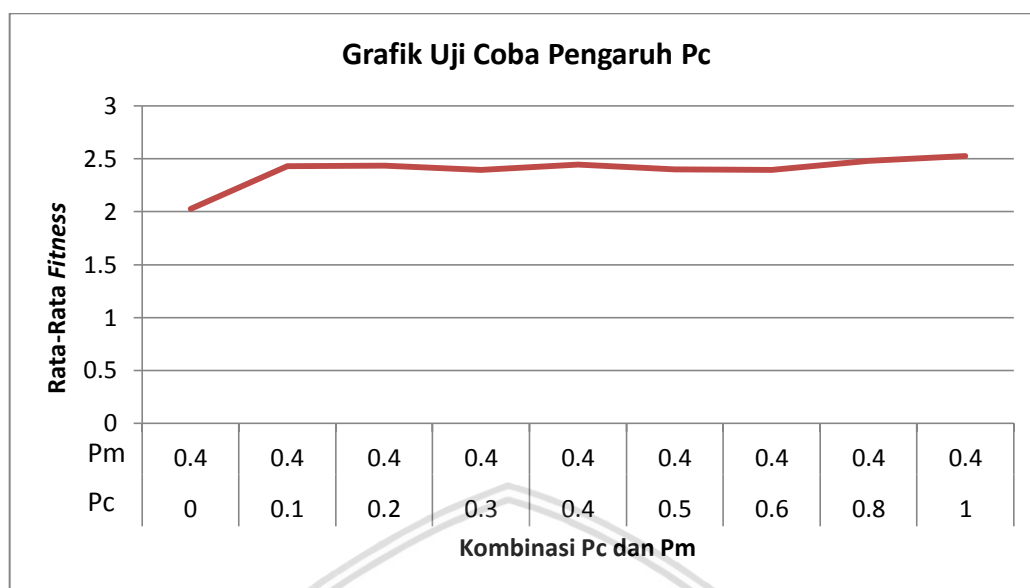
- Ukuran populasi : 500
- Banyaknya generasi : 100
- Probabilitas *crossover* : 0-1
- Probabilitas mutasi : 0.4

Pengujian dilakukan 10 kali untuk mendapatkan hasil yang mewakili kemampuan algoritme secara utuh. Nilai *fitness* pada setiap percobaan dihitung rata-ratanya untuk mengetahui pengaruh nilai P_c terhadap nilai *fitness*. Hasil uji coba pengaruh P_c dapat dilihat pada Tabel 6.6.

Tabel 6. 6 Hasil Uji Coba Pengaruh nilai P_c

Kombinasi		Nilai <i>Fitness</i>										Rata-Rata <i>Fitness</i>
		Percobaan Pengujian ke-										
Pc	Pm	1	2	3	4	5	6	7	8	9	10	
0	0.4	2.0397	2.0565	2.0134	2.1567	1.9581	2.0206	2.0053	2.0564	2.0065	1.9817	2.02949
0.1	0.4	2.5902	2.2062	2.5902	2.5351	2.4479	2.5351	2.5351	2.2328	2.5351	2.1043	2.43118
0.2	0.4	2.3024	2.5351	2.4672	2.5351	2.3024	2.5351	2.5902	2.3024	2.4672	2.3024	2.43395
0.3	0.4	2.0342	2.5902	2.2833	2.3024	2.5902	2.5351	2.5351	2.3024	2.5902	2.1755	2.39385
0.4	0.4	2.5351	2.5351	2.5902	2.3024	2.5902	2.3205	2.3008	2.5351	2.1732	2.5902	2.44726
0.5	0.4	2.3024	2.5902	2.5902	2.5902	2.2315	2.2197	2.5351	2.5351	2.221	2.1755	2.39907
0.6	0.4	2.1196	2.1755	2.5902	2.2197	2.5902	2.3024	2.3008	2.5351	2.5351	2.5902	2.39586
0.8	0.4	2.5351	2.3024	2.5902	2.5351	2.5255	2.5902	2.3024	2.5351	2.3024	2.5902	2.48085
1	0.4	2.5351	2.5902	2.5902	2.5351	2.5902	2.5351	2.5902	2.4868	2.3024	2.5351	2.52902

Berdasarkan Tabel 6.6 dapat diketahui nilai rata-rata *fitness* setiap percobaan kombinasi P_c 0-1 dan P_m 0.4. Untuk mempermudah melihat perbedaan rata-rata nilai *fitness* maka dibuatlah grafik rata-rata nilai *fitness* untuk uji coba pengaruh nilai P_c . Grafik pengaruh nilai P_c terhadap nilai *fitness* dapat dilihat pada Gambar 6.5.



Gambar 6. 5 Grafik Hasil Uji Coba Pengaruh Nilai Pc

Berdasarkan Gambar 6.5 dapat dilihat rata-rata nilai *fitness* pada kombinasi pc 0 dan pm 0.4 merupakan rata-rata *fitness* terendah dari pada kombinasi pc dan pm lainnya. Rata-rata nilai *fitness* pada kombinasi pc 0 dan pm 0.4 adalah 2.02949. Kemudian pada kombinasi pc 0.1 dan pm 0.4 rata-rata nilai *fitness* mengalami kenaikan dengan nilai 2.43118. Setelah kombinasi pc 0.1 dan pm 0.4 rata-rata nilai *fitness* cenderung stabil karena tidak mengalami perubahan yang signifikan. Rata-rata nilai *fitness* tertinggi terdapat pada kombinasi pc 1 dan pm 0.4 dengan nilai 2.52902. Berdasarkan tabel 6.5 nilai *fitness* terbesar yang didapatkan yaitu 2.5902 yang muncul pertama kali pada kombinasi pc 0.1 dan pm 0.4. Berdasarkan tabel 6.4 nilai *fitness* terbesar yang didapatkan yaitu 2.5902 yang muncul pada hasil uji coba kombinasi Pc lebih dari 0 dan Pm 0.4. Hal ini menunjukkan bahwa solusi terbaik akan muncul apabila probabilitas *crossover* bernilai lebih dari 0.

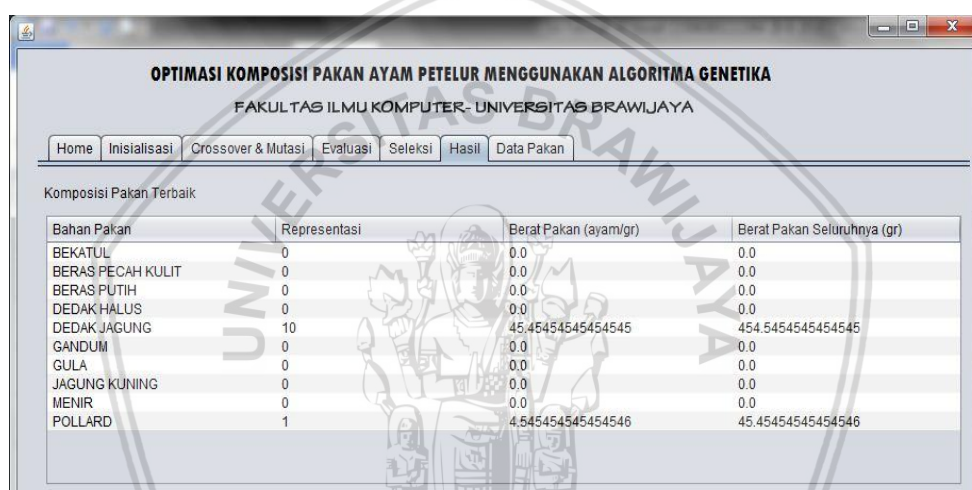
6.2.6 Pengujian dan Analisa Kasus

Pengujian kasus digunakan untuk membandingkan hasil komposisi pakan ayam petelur menggunakan algoritme genetika dengan pakan yang digunakan oleh peternak. Pengujian dilakukan pada ayam fase *starter* dengan kebutuhan pakan untuk satu ekor ayam yaitu 50 gram. Data pembanding diambil dari peternakan H.Suprpto yang terletak di Jl. Patimura no.6 rt 01 rw 06 Junrejo Batu. Untuk pakan ayam petelur yang digunakan pada peternakan H.Suprpto adalah pakan merk DMC 882 untuk fase ayam *starter* dengan umur 4-10 minggu. Harga pembelian pakan merk DMC 882 adalah Rp 450.000/sak dengan berat 1 sak sebesar 50 Kg. Jadi untuk satu ekor ayam membutuhkan biaya pakan Rp. 450. Untuk kandungan nutrisi yang terkandung pada pakan DMC 882 dapat dilihat pada Tabel 6.7.

Tabel 6. 7 Tabel Kandungan Nutrisi Pakan DMC 882

Nutrisi	Kandungan Nutrisi	Kandungan Nutrisi(1 ayam)
Energi Metabolisme (EM)	2.900 Kcal/Kg	145000 kal/gram
Protein Kasar (PK)	19%	9.5 gram
Serat Kasar (SK)	6.5%	3.3 gram
Phospor (P)	0.9%	0.5 gram

Berdasarkan pengujian generasi, populasi, kombinasi pc dan pm didapatkan nilai fitness terbaik yaitu 2.59017 dengan harga pakan yang didapatkan Rp. 381,818. Hasil komposisi pakan terbaik dapat dilihat pada Gambar 6.6.



Bahan Pakan	Representasi	Berat Pakan (ayam/gr)	Berat Pakan Seluruhnya (gr)
BEKATUL	0	0.0	0.0
BERAS PECAH KULIT	0	0.0	0.0
BERAS PUTIH	0	0.0	0.0
DEDAK HALUS	0	0.0	0.0
DEDAK JAGUNG	10	45.45454545454545	454.5454545454545
GANDUM	0	0.0	0.0
GULA	0	0.0	0.0
JAGUNG KUNING	0	0.0	0.0
MENIR	0	0.0	0.0
POLLARD	1	4.545454545454546	45.45454545454546

Gambar 6. 6 Komposisi Pakan Terbaik Ayam Petelur

Berdasarkan Gambar 6.6 didapatkan bahwa komposisi pakan terbaik yaitu dedak jagung 10 dan pollard 1. Untuk kandungan nutrisi dalam komposisi pakan terbaik dapat dilihat pada Tabel 6.8.

Tabel 6. 8 Tabel Kandungan Nutrisi Komposisi Pakan Terbaik

Nutrisi	Kandungan Nutrisi (1 ayam)
Energi Metabolisme (EM)	140000 kal/gram
Protein Kasar (PK)	5.5 gram
Serat Kasar (SK)	2.727272 gram
Phospor (P)	0.082727 gram
Klor (Cl)	0.035909 gram

Berdasarkan kebutuhan nutrisi ayam petelur fase *starter* pada Tabel 3.3 kebutuhan nutrisi ayam petelur fase *starter* dapat dilihat pada Tabel 6.9.

Tabel 6. 9 Tabel Kebutuhan Nutrisi Ayam Fase Starter

Nutrisi	Kandungan Nutrisi (1 ayam)
Energi Metabolisme (EM)	140000 kal/gram
Protein Kasar (PK)	9.5 gram
Serat Kasar (SK)	2.5 gram
Phospor (P)	0.4 gram
Klor (Cl)	0.075 gram

Untuk mempermudah melihat perbedaan hasil antara pakan hasil komposisi menggunakan algoritme genetika dengan kebutuhan nutrisi ayam petelur maka dibuatlah tabel perbandingan. Tabel perbandingan kandungan nutrisi hasil komposisi algoritme genetika dengan kebutuhan nutrisi dapat dilihat pada Tabel 6.10.

Tabel 6. 10 Tabel Perbandingan Kandungan Nutrisi

Pembanding	Kebutuhan Nutrisi	Kandungan Nutrisi Komposisi Algoritme Genetika	Terpenuhi(%)
Energi Metabolisme (EM)	140000 kal/gram	140000 kal/gram	100% terpenuhi
Protein Kasar (PK)	9.5 gram	5.5 gram	58% terpenuhi
Serat Kasar (SK)	2.5 gram	2.727272 gram	109% terpenuhi
Phospor (P)	0.4 gram	0.082727 gram	21% terpenuhi
Klor (Cl)	0.075 gram	0.035909 gram	48% terpenuhi
Harga	-	Rp 381,818	

Dari Tabel 6.10 dapat dilihat bahwa kandungan nutrisi hasil komposisi untuk energi metabolisme memenuhi 100% kebutuhan nutrisi ayam petelur. Pada protein kasar kandungan nutrisi terpenuhi sebesar 58%. Pada serat kasar kandungan nutrisi terpenuhi sebanyak 109%. Pada fosfor kandungan nutrisi terpenuhi sebanyak 21%. Sedangkan pada klor kandungan nutrisi terpenuhi sebanyak 48%. Berdasarkan data pakar nilai toleransi untuk pemenuhan nutrisi fase ayam starter yaitu $\pm 10\%$ untuk energi metabolisme dan protein karena pada fase ini nutrisi tersebut sangat penting untuk pertumbuhan ayam petelur. Untuk nutrisi serat kasar, fosfor dan klor nilai toleransi $\pm 15\%$. Dari Tabel 6.10 didapatkan kesimpulan bahwa energi metabolisme dan serat kasar masih dalam ambang batas toleransi. Sedangkan untuk protein kasar, fosfor, dan klor masih belum memenuhi kebutuhan nutrisi ayam petelur secara keseluruhan. Untuk harga pakan dengan hasil komposisi algoritme genetika masih lebih murah dibandingkan dengan harga pakan ayam petelur merk DMC.

BAB 7 PENUTUP

7.1 Kesimpulan

Berdasarkan hasil pengujian pada penelitian skripsi ini mengenai optimasi komposisi pakan ayam petelur menggunakan algoritme genetika, dapat disimpulkan sebagai berikut :

1. Algoritme genetika bisa diterapkan pada optimasi komposisi pakan ayam petelur dengan menggunakan representasi permutasi, *single-point crossover*, *reciprocal exchange mutation*, dan seleksi *elitism*.
2. Parameter masukan algoritme genetika yang terdiri dari iterasi, populasi, nilai Pc dan Pm mempengaruhi hasil nilai *fitness* yang dihasilkan. Pada percobaan generasi, semakin besar generasi maka nilai *fitness* akan semakin stabil dan akan terjadi konvergensi. Generasi dengan nilai *fitness* terbaik yaitu pada generasi 100 dengan nilai *fitness* 2.47973. Pada percobaan populasi, setelah populasi 300 rata-rata nilai *fitness* tidak mengalami kenaikan yang signifikan. Rata-rata nilai *fitness* tertinggi diperoleh pada populasi 500 dan 800 dengan rata-rata nilai *fitness* 2.573591. Pada pengujian kombinasi nilai Pc dan Pm diperoleh kesimpulan bahwa dengan nilai Pc 0 atau Pm 0 maka rata-rata nilai *fitness* akan mengalami penurunan. Rata-rata nilai *fitness* tertinggi diperoleh pada kombinasi Pc 0.5 dan Pm 0.3 dengan rata-rata nilai *fitness* 2.58459. Hasil uji coba kombinasi pc dan pm, pengaruh pm, dan pengaruh pc didapatkan kesimpulan bahwa solusi terbaik belum muncul saat nilai pc atau pm 0.
3. Hasil optimasi pakan ternak ayam petelur dengan nilai fitness terbaik yang dihasilkan menggunakan algoritme genetika dengan pilihan pakan bekatul, beras pecah kulit, beras putih, dedak halus, dedak jagung, gandum, gula, jagung kuning, menir dan pollard dengan komposisi 0,0,0,0,10,0,0,0,0,1, nilai fitness 2.59017 dan harga Rp. 381,818.
4. Dari hasil pengujian kasus didapatkan bahwa komposisi pakan dengan algoritme genetika mampu memenuhi kebutuhan nutrisi EM dan SK. Sedangkan untuk nutrisi PK, P dan Cl masih belum memenuhi kebutuhan nutrisi ayam petelur. Dari hasil pengujian kasus didapatkan bahwa harga pakan berdasarkan optimasi komposisi pakan dengan algoritme genetika lebih murah dibandingkan harga pembelian pakan peternak.

7.2 Saran

Berdasarkan penelitian yang dilakukan, Terdapat beberapa saran sebagai perbaikan untuk memperbaiki penelitian selanjutnya:

1. Penelitian ini dapat dikembangkan untuk menyelesaikan masalah optimasi komposisi pakan ternak ayam petelur dengan menambah kandungan nutrisi ayam petelur selain EM, PK, SK, Cl dan P.
2. Bahan pakan ayam petelur dapat dikembangkan lebih dari 50 bahan pakan sehingga mampu menghasilkan komposisi pakan yang lebih beragam.

3. Representasi kromosom, crossover, mutasi dan seleksi dapat menggunakan metode yang berbeda untuk mengetahui pengaruh perbedaan metode algoritme genetika terhadap nilai *fitness*.
4. Parameter pengujian kasus dapat dilakukan dengan kombinasi pakan selain bekatul, beras pecah kulit, beras putih, dedak halus, dedak jagung, gandum, gula, jagung kuning, menir dan pollard sehingga dapat ditemukan solusi yang lebih baik dari solusi yang sudah ditemukan.



DAFTAR PUSTAKA

- Abidin, Ir.Zainal.2004."Meningkatkan Produktivitas Ayam Ras Petelur". PT.Agromedia Pustaka.Depok
- Abun.2006.*Protein dan Asam Amino Pada Unggas*.Jurusan Nutrisi dan Pakan Ternak, Fakultas Peternakan, Universitas Padjajaran. Jatinagor.
- Aksi Agraris Kanisius. 2003. *Beternak Ayam*. Cetakan ke-21. Penerbit Kanisius, Yogyakarta.
- Andrianto, Muflikhah, Rahayudi. 2017. "*Optimasi Komposisi Pakan Kuda Dewasa Menggunakan Algoritma Genetika*". Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya.
- Ariwibowo, Arnold,dkk.2008.*Penerapan Algoritme Genetika Pada Penentuan Komposisi Pakan Ayam Petelur*. Seminar Nasional Aplikasi Teknologi Informasi.Yogyakarta.
- Defersha, F & Chen, M 2010, "A Parallel Genetic Algorithm For A Flexible Job-Shop Scheduling Problem With Sequence Dependent Setups". The International Journal of Advanced Manufacturing Technology, vol. 49, no. 1, pp. 263-279.
- Drh. Zulfikar, M.Si.2013."*Managemen Pemeliharaan Ayam Petelur Ras*". Peternakan dan Praktisi Kesehatan Hewan, Fakultas Pertanian, Universitas Syiah Kuala.Aceh.
- Gen, Mtsuo and Cheng, R. (1997). Genetic Algorithm and Engineering Design. John Wiley & Son, Inc. USA
- Insani, Galuh Adi.2009.*Lemak dan Fungsinya Pada Ayam*. [online] tersedia di: <<https://chickaholic.wordpress.com/2007/11/06/lemak-dan-fungsinya-pada-ransum-ayam/>> [Diakses pada 20 Desember 2015]
- Japfa. 2004. *Pedoman Pemeliharaan Layer MB 402*.PT Multibreeder Adimira Indonesia.Jakarta.
- Kementerian Pertanian, Sekretariat Jenderal, Pusat Data Dan Sistem Informasi Pertanian.2017.*Outlook Telur*. [pdf] tersedia di: <http://epublikasi.setjen.pertanian.go.id/download/file/395-outlook-telur-2017> [Diakses 15 April 2018]
- Kusumadewi, Sri. 2003. *Artificial Intelligence (Teknik dan Aplikasinya)*.Graha Ilmu, Yogyakarta.
- Madja. 2007.*Lemak Dalam Tubuh*. [online] tersedia di: <<http://madja.wordpress.com/lemak-dalam-tubuh/>> [Diakses 23 Desember 2015].

- Mahmudy, WF, Marian, RM & Luong, LHS 2012a, 'Solving part type selection and loading problem in flexible manufacturing system using real coded genetic algorithms – Part II: optimization', *International Conference on Control, Automation and Robotics*, Singapore, 12-14 September, World Academy of Science, Engineering and Technology, pp. 706-710.
- Mahmudy, WF, Marian, RM & Luong (2014), 'Hybrid genetic algorithms for part type selection and machine loading problems with alternative production plans in flexible manufacturing system', *ECTI Transactions on Computer and Information Technology (ECTI-CIT)*, vol. 8, no. 1, pp. 80-93.
- Mahmudy, WF & Rahman, MA 2011, "*Optimasi Fungsi Multi-Obyektif Berkendala Menggunakan Algoritma Genetika Adaptif Dengan Pengkodean Real*". *Kursor*, vol. 6, no. 1, pp. 19-26.
- Mahmudy, WF.2013. "*Algoritme Evolusi*". Program Teknologi Informasi dan Ilmu Komputer, Universitas Brawijaya. Malang
- Marginingtyas, Ervina, Mahmudy, WF, Indriarti. 2015. "*Penentuan Komposisi Pakan Ternak Untuk Memenuhi Kebutuhan Nutrisi Ayam Petelur Dengan Biaya Minimum Menggunakan Algoritme Genetika*". *Teknik Informatika, FILKOM*, Universitas Brawijaya. Malang.
- Marian, RM 2003, "*Optimisation Of Assembly Sequences Using Genetic Algorithms*". School of Advanced Manufacturing and Mechanical Engineering. University of South Australia.
- Menteri Negara Riset dan Teknologi Bidang Pendayagunaan dan Pemasyarakatan Iptek (Menegristek). 2000. "*Budidaya Ayam Petelur*". Menteri Negara Riset dan Teknologi Bidang Pendayagunaan dan Pemasyarakatan Iptek.
- Michalewicz, Zbigniew. (1996). *Genetic Algorithm + Data Structure = Evolution Programs*. Springer-Verlag 3rd Edition.
- Murtidjo, Bambang Agus. 2006. "*Pedoman Meramu Pakan Unggas*". Cetakan ke-17. Penerbit Kanisius, Yogyakarta.
- PENS. Modul Bab 7 Algoritme Genetika. [pdf] tersedia di: <http://entin.lecturer.pens.ac.id/Kecerdasan%20Buatan/Buku/Bab%207%20Algoritme%20Genetika.pdf> [Diakses pada 5 Januari 2015]
- Poli, Riccardo, Langdon, W.B. 1997. *Genetic Programming with One-Point Crossover and Point Mutation*. School of Computer Science, The University of Birmingham. United Kingdom.
- Rasyaf, Dr. Ir. Muhammad. 2001. "*Beternak Ayam petelur*". Cetakan ke-15. Penerbit Kanisius, Yogyakarta.

- Rasyaf, M. 1992a. *Produksi dan Pemberian Ransum Unggas*. Penerbit Kanisius, Yogyakarta.
- Rismawan, Tedy ; Kusumadewi, Sri. (2007). "*Rancang Bangun Sistem Penentuan Komposisi Bahan Pangan Harian Menggunakan Algoritme Genetika*". Teknik Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia. Yogyakarta.
- Suyanto. 2010. "*Algoritma Optimasi; Deterministik atau Probabilistik*". Graha Ilmu Yogyakarta.
- Wardhani ,Luh Kesuma, M. Safrizal, Chairi, Ahmad. 2011."*Optimasi Komposisi Bahan Pakan Ikan Air Tawar menggunakan Metode Multi-Objective genetic Algorithm*". Teknik Informatika, Fakultas Sains dan teknologi, Universitas Islam Negeri Sultan Syarif Kasim. Riau.
- Warintek.2004.*Budidaya Ayam Petelur*.pdf tersedia di:
<http://www.warintek.ristek.go.id/peternakan/budidaya/ayam_petelur.pdf> [Diakses 20 Desember 2015]
- Winarti, Wiwi, Dkk.2006.*Kimia untuk SMA/MA Kelas XII*.pdf tersedia di:
<<http://ebook.sman1-slo.sch.id/Kimia/KIMIA%20XII%20PS/PDF/>>[Diakses 21 Desember 2015]
- Yogeswaran, M, Ponnambalam, SG & Tiwari,MK.2009.*An efficient hybrid evolutionary heuristic using genetic algorithm and simulated annealing algorithm to solve machine loading problem in FMS*.International Journal of Production Research, vol. 47, no. 19, pp. 5421-5448.m